

# **Internationales Steuerinformationssystem**

- Auf Basis des European Tax Analyzer -

Inauguraldissertation  
zur Erlangung der Würde eines  
Doktors der Wirtschaftswissenschaften  
der Universität Mannheim

vorgelegt von  
Diplom-Wirtschaftsinformatiker Michael Grünewald  
aus Ludwigshafen

Dekan:	Dr. Jürgen M. Schneider
Referent:	Prof. Dr. Christoph Spengel
Korreferent:	Prof. Dr. Armin Heinzl
Tag der mündlichen Prüfung:	6. Oktober 2010

## Inhaltsverzeichnis

1	Einführung	1
1.1	Problemstellung	1
1.2	Zielsetzung	4
1.3	Vorgehensweise	5
2	European Tax Analyzer	7
2.1	Überblick	7
2.2	Beschreibung	10
2.2.1	Benötigte Informationen und bisheriges Datenmodell	10
2.2.2	Berechnung der Steuerbelastung	20
3	Anforderungen	25
3.1	Speicherung	25
3.2	Vereinfachte Zeitreihenanalyse	27
3.2.1	Dokumentation der Steuersysteme	27
3.2.2	Recherche	28
3.2.3	Statusinformationen und Datenqualität	29
3.2.3.1	Benutzerverwaltung	29
3.2.3.2	Versionshistorie	29
3.2.4	Kontrollfunktion	30
3.3	Nebenbedingungen	30
3.3.1	Verwendung ohne Anbindung an die Datenbank	30
3.3.2	Performanz	30
3.4	Zusammenfassung	31
4	Evaluation geeigneter Methoden zur Umsetzung	32
4.1	Auswahl der geeigneten Notation bzw. des Datenformates	32
4.1.1	Bisheriges Datenmodell des European Tax Analyzer	32

4.1.2	Aktuelles Konzept der Datenspeicherung	34
4.2	Mehrbenutzerbetrieb	37
4.3	Anforderungen an das Datenbankschema	39
4.4	Performanz	40
4.5	Dokumentation der Steuersysteme	41
4.6	Objektorientierte vs. relationale Datenbank	42
4.7	Weboberfläche	46
4.8	Refaktorisierung & Reverse Engineering oder Neuentwurf	48
4.9	Data Access Objects (DAO)	49
4.10	Sicherheit	50
4.11	Ergebnisse	51
5	Umsetzung der Anforderungen mithilfe ausgesuchter Methoden	53
5.1	Transaktionsverwaltung	53
5.1.1	Lange versus kurze Konversationen	54
5.1.2	Sperrstrategie	56
5.2	Kopierinfrastruktur	58
5.2.1	XML-Encoder	59
5.2.2	Datenbank: ID und Versionsattribute	60
5.3	Modifikationen am Datenmodell	60
5.3.1	Überblick	60
5.3.2	Anpassungen des Ländermodells	64
5.3.3	Zerlegung des Unternehmensmodells	66
5.3.4	Herstellungskostenermittlung & Verbrauchsfolge	70
5.3.5	Abschreibungen	76
5.3.6	Betriebliche Altersversorgung	80
5.3.7	Steuern	83



5.3.7.1	Modellierung des Steuerartensystems	83
5.3.7.2	Bemessungsgrundlagen	86
5.3.7.2.1	Sammelbaustein und Tarif	86
5.3.7.2.2	Andere Steuern	91
5.3.7.2.3	Verlustverrechnungsmodul	93
5.3.7.2.4	Formelbasierte Bemessungsgrundlagen	100
5.3.7.2.4.1	Grundlagen	100
5.3.7.2.4.2	Bilanz	106
5.3.7.2.4.3	Gewinn-und-Verlust-Rechnung	108
5.3.7.2.4.4	Anteilseigner	109
5.4	PDF-Verwaltung	111
5.5	Qualitäts- und Benutzerverwaltung	113
5.6	Finales Datenbankschema	116
5.6.1	Übersicht	116
5.6.2	Besonderheiten	118
5.6.3	Formale Anforderungen	119
5.7	Sonstige Modifikationen	125
5.7.1	Ereignissystem für Menüelement	125
5.7.2	Zentralisierte Speicherung	126
5.7.3	Tabellen für Ausgabemasken	128
6	Fallbeispiel	130
6.1	Sachverhalt	130
6.2	Benutzer anlegen	131
6.3	Ökonomische Daten	133
6.4	Erstellen des Steuersystems für 1990	136
6.5	Folgende Jahre	151

6.6	Dokumentation hinzufügen	155
6.7	Speichern und Laden	160
6.8	Berechnen einer Zeitreihe	163
6.9	Recherche und Kontrolle im Webinterface	167
6.9.1	Länderdetails	167
6.9.2	Erstellen und Anzeige von Views	168
6.9.3	Kontrolle	176
6.9.4	Suche nach Steuern und Dokumenten	177
6.10	Wechsel des ökonomischen Rahmenmodells	180
6.10.1	Branchenwechsel	180
6.10.2	Wechsel der Unternehmensgröße	183
6.10.3	Finanzkrise	184
6.11	Sicherung (Backup) der Datenbank	187
7	Zusammenfassung	189
	Literaturverzeichnis	191

## Abbildungsverzeichnis

Abbildung 1: Interaktion zwischen Unternehmensmodell und Steuern	10
Abbildung 2: Ländermodell, Unternehmensmodell und Steuerdaten	12
Abbildung 3: Vererbungshierarchie der Bilanzobjekte	14
Abbildung 4: Implementierung der Abschreibungen	16
Abbildung 5: Lagerbestandsverwaltung und Verbrauchsfolge	17
Abbildung 6: Klassen für die Ermittlung der Herstellungskosten	17
Abbildung 7: Vererbungshierarchie der Bemessungsgrundlagen	18
Abbildung 8: Steuerdaten, Steuerobjekte und Bemessungsgrundlagen	19
Abbildung 9: Implementierung der Steuertarife	20
Abbildung 10: “Vereinfachtes Ablaufschema der Modellrechnungen“	24
Abbildung 11: Standard Use Cases	54
Abbildung 12: UML-Diagramm des UnternehmensmodellVst	63
Abbildung 13: ID des Ländermodells	65
Abbildung 14: Unternehmensmodell nach Zerlegung	67
Abbildung 15: Übersicht über wichtige Aufzählungen zur Verknüpfung von ökonomischen und steuerlichen Daten	68
Abbildung 16: Auflösung der indirekten Verknüpfungen über das Unternehmensmodell	69
Abbildung 17: Dialog zur Eingabe der Regeln zur Ermittlung der Herstellungskosten	71
Abbildung 18: Bewertungsobjekt und Berechnungsmodule für die Herstellungskosten	71
Abbildung 19: Liste aller möglichen Bestandteile der Herstellungskosten und der benutzten Infrastruktur	73
Abbildung 20: Klassendiagramm zur Berechnung des Lagerbestandes	74
Abbildung 21: Eingabedialog der Universalmethode	77
Abbildung 22: Auswahldialog der Abschreibungsmethoden	78
Abbildung 23: Vererbungshierarchie der verschiedenen Abschreibungsmethoden	78
Abbildung 24: Personalverwaltung des Variable International Tax Analyzer	81
Abbildung 25: Datenmodell der betrieblichen Altersvorsorge	82
Abbildung 26: Polymorphismus der Berechnungsmodule	83
Abbildung 27: Metamodell zur Modellierung des Steuerartensystems	84
Abbildung 28: Dialog zum Bearbeiten von Steuern	85
Abbildung 29: Benutzeroberfläche des Sammelbausteins (inkl. Steuerbaum links)	87

Abbildung 30: Zusammenspiel zwischen Steuerobjekt, TaxBaseSammelbaustein und BMG_Baustein	88
Abbildung 31: Implementierung des Tarifs	89
Abbildung 32: Eingabemaske des Universaltarifs	90
Abbildung 33: Klassendiagramm andere Steuern	91
Abbildung 34: Benutzeroberfläche zum Rückgriff auf andere Steuern	92
Abbildung 35: Fehlermeldung beim Löschen einer Steuer	93
Abbildung 36: GUI des Verlustmoduls für den Vortrag	95
Abbildung 37: GUI des Verlustmoduls für den Rücktrag	97
Abbildung 38: Datenmodell und Implementierung der Verlustverrechnung	99
Abbildung 39: Infrastruktur zur Auflösung von Variablen	102
Abbildung 40: Infrastruktur zur Unterstützung der Baumstruktur der formelbasierten Bemessungsgrundlagen	104
Abbildung 41: Dialog-Hierarchie für formelbasierte Bemessungsgrundlagen	105
Abbildung 42: Benutzeroberfläche für die Bemessungsgrundlage Bilanz	107
Abbildung 43: Benutzeroberfläche für die Gewinn-und-Verlust-Rechnung	108
Abbildung 44: Eingabemaske Anteilseigner Vermögensgegenstände	110
Abbildung 45: Eingabemaske Anteilseigner Einkunftsarten	110
Abbildung 46: PDF-Verwaltung	112
Abbildung 47: Benutzer- und Qualitätsverwaltung	115
Abbildung 48: Finales Datenbankschema -- Übersicht	122
Abbildung 49: Teil des Datenbankschemas I/II -- Detailansicht	123
Abbildung 50: Teil des Datenbankschemas II/II -- Detailansicht	124
Abbildung 51: Ereignissystem der Menüstruktur und definierte Ereignisse	125
Abbildung 52: Zentrale Infrastruktur zum Speichern und Laden von XML-Dateien	127
Abbildung 53: Tablemodels für die Datenausgabe	129
Abbildung 54: Anlegen eines Benutzers	131
Abbildung 55: Beispieltabelle für Benutzerdaten	131
Abbildung 56: Benutzeranmeldung	132
Abbildung 57: Dialog zum Wechseln des Passwortes	132
Abbildung 58: Eingabe der Anteilseignerdaten	133
Abbildung 59: Eingabe der Bilanzdaten	134
Abbildung 60: Eingabe der Details der einzelnen Bilanzpositionen	134

Abbildung 61: Eingabe der Inflationsraten und Zinsen	135
Abbildung 62: Eingabe des Unternehmensplans	136
Abbildung 63: Erzeugen eines neuen Ländermodell	137
Abbildung 64: Auswahl der Gewährleistungsrückstellung	137
Abbildung 65: Eingabe der Durchführungsdetails für die Betriebliche Altersvorsorge	138
Abbildung 66: Eingabe der Abschreibungen	139
Abbildung 67: Eingabe des Verbrauchsfolgeverfahrens	139
Abbildung 68: Einbezug der Grundsteuer in die Herstellungskosten	140
Abbildung 69: Eingabe der Steuern	140
Abbildung 70: Auswahl der Berechnungsperioden einer Steuer	141
Abbildung 71: Auswahl der Steuerart	141
Abbildung 72: Einbezug einer Steuer in die HK ermöglichen und deren Aufteilung festlegen	142
Abbildung 73: Bemessungsgrundlage Grundsteuer	142
Abbildung 74: Ermittlung des Betriebsvermögens für die Vermögensteuer 1990	143
Abbildung 75: Rückgriff auf Bilanzdaten zur Ermittlung des Einheitswertes für die Grundsteuer	144
Abbildung 76: Eingabe des Tarifs der Grundsteuer	145
Abbildung 77: Abzug anderer Steuern in der Körperschaftsteuer	146
Abbildung 78: Zuordnung der Steuersysteme zu den Anteilseignern	147
Abbildung 79: Kirchensteuertarif	148
Abbildung 80: Einkommensteuertarif 1990	149
Abbildung 81: Steuern des Unternehmens für 1990 im Ausgangsfall	150
Abbildung 82: Steuern der Anteilseigner für 1990 im Ausgangsfall	151
Abbildung 83: Landänderungsdialog nach erneutem Laden	152
Abbildung 84: Erkennung vorhandener weiterführender Dokumentation auf Länderebene	156
Abbildung 85: Tarif der Abgeltungsteuer	157
Abbildung 86: Erkennung und Auswahl vorhandener weiterführender Dokumentation auf Steuerebene	158
Abbildung 87: Verwaltung von Dokumenten im Webinterface	159
Abbildung 88: Auswahldialog zum Laden aus der Datenbank	160
Abbildung 89: Simulationsmenü	161
Abbildung 90: Datenbankmenü	161

Abbildung 91: Dialogabfolge beim Speichern in die Datenbank	162
Abbildung 92: Steuerbelastung in Deutschland auf Unternehmensebene im Zeitablauf	164
Abbildung 93: Steuerbelastung in Deutschland auf Anteilseignerebene im Zeitablauf	166
Abbildung 94: Steuerbelastung in Deutschland auf Gesamtebene im Zeitablauf	167
Abbildung 95: Recherche mithilfe der Baumstruktur	168
Abbildung 96: Menü zur Auswahl und Eingabe von SQL-Abfragen	169
Abbildung 97: Eingabe einer SQL-Abfrage	170
Abbildung 98: Ausgabe der Grundsteuertarife im Webinterface	171
Abbildung 99: Körperschaftsteuertarife für Deutschland von 1990-2009	172
Abbildung 100: Tarif der Gewerbesteuer vom Ertrag	172
Abbildung 101: Tarife der Gewerbesteuer vom Kapital	173
Abbildung 102: Zeitreihe der Einkommensteuertarife (nur Änderungsjahre)	174
Abbildung 103: Tarife der Vermögensteuer auf Unternehmensebene	175
Abbildung 104: Vermögensteuertarife für Anteilseigner	175
Abbildung 105: Tarifsätze für den Solidaritätszuschlag	176
Abbildung 106: Ausschnitt der kompakten Länderübersicht	177
Abbildung 107: Suche nach Steuern	178
Abbildung 108: Resultat der Suche	178
Abbildung 109: Suchmaske für PDF-Dateien	179
Abbildung 110: Resultat der PDF-Suche	179
Abbildung 111: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Unternehmen)	181
Abbildung 112: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Anteilseigner)	182
Abbildung 113: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Gesamtebene)	182
Abbildung 114: Verarbeitendes Gewerbe groß (Unternehmensebene)	183
Abbildung 115: Verarbeitendes Gewerbe groß (Ebene der Eigner)	184
Abbildung 116: Verarbeitendes Gewerbe groß (Gesamt)	184
Abbildung 117: Steuerbelastung in Deutschland im Zeitablauf (Finanzkrise - Unternehmensebene)	185
Abbildung 118: Steuerbelastung in Deutschland auf Anteilseignerebene im Zeitablauf	186

Abbildung 119: Steuerbelastung in Deutschland auf Gesamtebene im Zeitablauf

(Finanzkrise)

187

Abbildung 120: Wechsel der Datenbankanbindung

188

**Tabellenverzeichnis**

Tabelle 1: Vermögens- und Kapitalausstattung (Bilanz)	14
Tabelle 2: Beispiel zur Entstehung von stillen Lasten und Reserven (Angaben in €)	22
Tabelle 3: Verlustverrechnungsbeispiel	96
Tabelle 4. Auflistung der relevanten Steuerrechtsänderungen von 1991-2009	152
Tabelle 5: Steuerbilanz des Unternehmens in Periode 6	163
Tabelle 6: Kennzahlen der Modellunternehmen	163
Tabelle 7: Kennzahlen für die Energiebranche und das Verarbeitende Gewerbe (groß)	180



**Abkürzungsverzeichnis:**

Abb.	Abbildung
AE	Anteilseigner
AHK	Anschaffungs- und Herstellungskosten
AS	Aufteilungsschlüssel
BewG	Bewertungsgesetz
BMG	Bemessungsgrundlage
bzw.	beziehungsweise
CSS	Cascade Style Sheets
d. h.	das heißt
DAO	Data Access Object
DBA	Doppelbesteuerungsabkommen
DBMS	Database Management System
ESt	Einkommensteuer
EStG	Einkommensteuergesetz
EStR	Einkommensteuerrichtlinien
etc.	et cetera
EU	Europäische Union
f. (ff.)	folgende (fortfolgende)
FIFO	First in - First Out
GA	Gesamtaufwand

GEW .....	Gemeiner Wert
GewSt .....	Gewerbesteuer
GewStG .....	Gewerbesteuergesetz
GGF .....	Gesellschafter-Geschäftsführer
ggü. ....	gegenüber
GrStG .....	Grundsteuergesetz
GUI .....	Graphical User Interface
GuV .....	Gewinn-und-Verlust-Rechnung
HIFO .....	Highest in - First out
Hrsg. ....	Herausgeber
HTML .....	Hypertext Markup Language
HTTP .....	Hypertext Transfer Protocol
i. A. ....	im Allgemeinen
i. V. m. ....	in Verbindung mit
IAS .....	International Accounting Standards
IBFD .....	International Bureau of Fiscal Documentation
ID .....	Identifier
IFRS .....	International Financial Reporting Standards
IT .....	Informationstechnologie
J2SE .....	Java 2 Platform, Standard Edition
JDBC .....	Java Database Connectivity
JDK .....	Java Developer Kit
JEP .....	Java Mathematical Expression Parser

JSF.....JavaServer Faces

KSt..... Körperschaftsteuer

KStG..... Körperschaftsteuergesetz

LIFO .....Last in - First out

LVA..... Lettland

m. w. V..... mit weiteren Verweisen

m. E. .... meines Erachtens

MIME..... Multipurpose Internet Mail Extensions

OECD ..... Organization for Economic Cooperation and Development

PDF.....Portable Document Format

POJO ..... Plain Old Java Object

RBW ..... Restbuchwert

RBWA..... Restbuchwert zu Beginn der aktuellen Periode

RBWE ..... Restbuchwert am Ende der aktuellen Periode

S..... Seite

SA.....Steuerart

SAS..... Steuerartensystem

SM.....Steuerm modul

SQL ..... Structured Query Language

SVR..... Sachverständigenrat

u. a. ....	unter anderem
UML.....	Unified Modeling Language
URL.....	Uniform Resource Locator
vgl. ....	vergleiche
VITAX .....	Variable International Tax Analyzer
WBK.....	Wiederbeschaffungskosten
XML.....	Extensible Markup Language
z. B. ....	zum Beispiel
ZEW .....	Zentrum für Europäische Wirtschaftsforschung
ZWE .....	Zeitwert
ZWEA .....	Zeitwert zu Beginn der aktuellen Periode
ZWEE.....	Zeitwert am Ende der aktuellen Periode

## **1 Einführung**

### **1.1 Problemstellung**

Steuerbelastungsvergleiche in der betriebswirtschaftlichen Steuerlehre dienen dazu, die Wirkungen von Steuersystemen zu beurteilen. Zum einen sollen Steuerbelastungsvergleiche Entscheidungshilfen für steuerplanerische Fragestellungen liefern, zum anderen sollen Steuersysteme hinsichtlich ihrer Entscheidungsneutralität und Allokationseffizienz beurteilt werden. Eine Computerunterstützung von Steuerbelastungsvergleichen soll dem Anwender bei den komplexen Berechnungen assistieren.

Der European Tax Analyzer,<sup>1</sup> der an der Universität Mannheim in Kooperation mit dem Zentrum für Europäische Wirtschaftsforschung (ZEW) entwickelt wurde, hat sich als eines der führenden quantitativen, einzelfallbezogenen Analyseinstrumente für Steuerbelastungsvergleiche etabliert.

Er simuliert die Entwicklung eines Unternehmens und seine Besteuerung über zehn Perioden. Als Vergleichsmaßstab dient die Entwicklung der Gesellschaft ohne jegliche Steuern. Die Differenz zu dem durch Steuerzahlungen geminderten Endvermögen ergibt, vereinfacht ausgedrückt, die effektive Steuerbelastung. Dadurch sind Standortunterschiede aufgrund steuerlicher Regelungen unter Einbeziehung einer Vielzahl von Ermittlungsfaktoren berechenbar.

In der ersten Version wurden zunächst drei Länder<sup>2</sup> betrachtet. Damals waren die Steuersysteme noch direkt im Quellcode hinterlegt. Aufgrund des hohen Programmieraufwandes, zum Anlegen und Pflegen, stieg die Anzahl der Länder nur noch auf insgesamt fünf an. Durch eine Neuimplementierung, als „Variable International Tax Analyzer (VITAX)“<sup>3</sup> bezeichnet, wurde dies grundlegend geändert. Hierdurch wurde eine deutliche Erhöhung der Anzahl an Ländern ermöglicht.

Mittlerweile werden sämtliche Staaten der Europäischen Union sowie einige ausgewählte der Organisation für wirtschaftliche Zusammenarbeit und Entwicklung

---

<sup>1</sup> Vgl. Spengel, C., Steuerbelastungsvergleiche, 1995; Jacobs, O. H./Spengel, C., European Tax Analyzer, 1996; Meyer, R., Simulation von Steuerbelastungen, 1996.

<sup>2</sup> Dies waren Deutschland, Frankreich und Großbritannien. Vgl. Spengel, C., Steuerbelastungsvergleiche, 1995.

<sup>3</sup> Stetter, T., VITAX, 2005, S. 3.

(OECD), ständig aktualisiert. Des Weiteren wurden unlängst Länder aus dem asiatischen Raum einbezogen.

Neuere Entwicklungen, wie z. B. das ZEW Corporate Taxation Microsimulation Model (ZEW TaxCoMM)<sup>4</sup>, eine Mikrosimulation auf der Basis von Handelsbilanzen von deutschen Kapitalgesellschaften, zielen primär auf die Analyse von Aufkommens- und Verteilungswirkungen der Besteuerung ab. Aufgrund dessen ist nicht zu erwarten, dass der European Tax Analyzer in naher Zukunft an Bedeutung verlieren wird, da beide Ansätze komplementäre Analysespektren bedienen.

Zur Messung von Steuerbelastungen und zur Beurteilung von Reformvorschlägen wird am Zentrum für Europäische Wirtschaftsforschung (ZEW) eine umfangreiche Datenbasis benötigt. In einem kürzlich abgeschlossenen Forschungsprojekt<sup>5</sup> wurde detailliertes Wissen über die Steuersysteme von allen Mitgliedsländern der Europäischen Union und für fünf Länder sogar verschiedene Rechtsstände benötigt. Durch die Variationen, die zur Analyse verschiedener Optionen zur Ausgestaltung einer harmonisierten Körperschaftsteuerbemessungsgrundlage (z. B. Abschreibungsregelungen oder Vorschriften zur Vorratsbewertung) notwendig waren, vervielfachte sich die Anzahl immens.<sup>6</sup> Jede Änderung bei den steuerlichen Regelungen wird im jetzigen Datenmodell als eigenes Land betrachtet, sodass insgesamt zweihundertsechsfundfünfzig verschiedene Instanzen vorhanden sind. Zudem wurden insgesamt sieben verschiedene Branchen, sowohl für kleine und mittelständische als auch große Unternehmen angelegt. Des Weiteren erstreckte sich die Analyse über verschiedene Modifikationen der ökonomischen Ausgangsdaten. So wurde die Anlagenintensität, Personalintensität, Umsatzrentabilität und die Personalintensität in mehreren Schritten variiert. Diese gewaltigen Datenmengen führten dazu, dass vier wissenschaftliche Mitarbeiter, etliche Praktikanten und Hilfskräfte für mehrere Monate mit der Bearbeitung dieses Projektes beschäftigt waren. Durch die hohe Anzahl an beteiligten Personen und die Fülle an zu verarbeitenden Informationen, hat sich deutlich gezeigt, dass es dringend einer Infrastruktur bedarf, die in der Lage ist, die Koordination und Synchronisation der Modifikationen zu unterstützen.

---

<sup>4</sup> Vgl. Reister, T., Mikrosimulationsmodelle, 2009.

<sup>5</sup> Project for the EU Commission TAXUD/2007/DE/325.

<sup>6</sup> Vgl. Oestreicher, A., et al., WTJ 2009, S. 46-66

Von den Auftraggebern werden in zunehmendem Maße, neben den jeweils aktuellen Rechtständen einzelner Länder, auch historische Daten verlangt. Zur Berechnung von Zeitreihen ist es notwendig, ganze Zeiträume abdecken zu können. Durch die Häufigkeit von Änderungen im Steuerrecht bleibt ein Rechtsstand nur selten über einen längeren Zeitraum hinweg gültig, sodass für jedes Jahr eine separate Instanz benötigt wird. Diese Änderungen (z. B. Senkung des Steuersatzes) oder Variationen zur Berechnung von Reformoptionen betreffen meist nur einen kleinen Teil der bereits vorhandenen Modellierungen, so dass eine effiziente Option der Wiederverwendung sinnvoll erscheint. Jedoch birgt dies auch das Risiko, dass sich Fehler über verschiedene Instanzen hinweg fortpflanzen, sodass bei der Entdeckung einer solchen Inkorrektheit zumindest die Möglichkeit bestehen sollte, den Weg zur Quelle nachverfolgen zu können.

Der European Tax Analyzer existiert bereits schon seit über einem Jahrzehnt, sodass zumindest für einzelne Länder diese Daten bereits vorliegen müssten. In der Praxis hat sich jedoch gezeigt, dass durch die ständige Fortentwicklung dieses Werkzeuges meistens auch die bisherigen Datenbestände unbrauchbar wurden. Selbst wenn diese Daten noch verfügbar wären, wäre der notwendige Aufwand zur Aufbereitung wohl recht hoch.

Des Weiteren lag eine Dokumentation im Ermessen des jeweiligen Mitarbeiters und häufig nur diesem vor. Zudem müssen diese Informationen für das Programm aufbereitet werden, was aufgrund von verschiedenen Bearbeitern häufig recht unterschiedlich ausfällt. Aus diesem Grund wurden die benötigten Informationen nicht selten mehrfach recherchiert. Ein zentrales, für jeden Mitarbeiter einfach zugängliches Archiv, das solche Redundanzen vermeiden könnte, gibt es bisher nicht. Ebenso bleibt häufig unklar, in wieweit diese Daten bereits validiert wurden oder welchen Zeitpunkt eines Rechtsstandes sie abbilden. Durch die stark angestiegene Anzahl der gepflegten Länder ist absehbar, dass der jetzige Zustand bald nicht mehr tragbar sein wird.

Der European Tax Analyzer ist als Einzelplatzanwendung konzipiert, so dass die Notwendigkeit einer Synchronisierung der benötigten Daten, z. B. über ein Daten-

bankmanagementsystem, bisher nicht vorgesehen ist.<sup>7</sup> Die Länder werden meistens nur projektabhängig betreut und nicht zentral verwaltet, sodass es häufig vorkommt, dass mehrere Versionen des gleichen Landes für das gleiche Jahr existierten und der abgebildete Rechtsstand (z. B. ob Reformvorschläge oder Steueränderungen des nächsten Jahres bereits mit einbezogen wurden) ungewiss war. Dadurch herrschte oft Unklarheit über die Qualität und Aktualität einer Modellierung, sodass immer wieder erneuter Aufwand notwendig war, dies zu überprüfen und sicherzustellen. Des Weiteren werden wenig nachgefragte Länder (z. B. Kanada) nur bei Bedarf aktualisiert, sodass diese nur noch in veralteten Datenbeständen vorhanden sind. Die notwendige Dokumentation wird bisher in Papierform in Ordnern gesammelt und enthält wenig direkten Bezug zur jeweiligen konkreten Modellierung bzw. Umsetzung.

## 1.2 Zielsetzung

Mit der wachsenden Anzahl der Länder hat sich die aktuelle, versionsabhängige Speicherung und die nicht einheitliche Dokumentation der Steuersysteme als problematisch erwiesen. Zudem erfordert die erhöhte Anzahl gleichzeitiger Bearbeiter eine unterstützende Infrastruktur für die Synchronisation von Modifikation und Verwaltung der gemeinsamen Datenbasis. Diese sollte es zudem ermöglichen, nahezu alle vorhandenen Quellen, wenn möglich kontextbezogen, zu archivieren. Zur Sicherstellung der Qualität erscheint es sinnvoll, aktuelle Rechtsstände von jeglichen Variationen zu trennen und einzelnen Nutzern die Gelegenheit zu geben, die Güte einer solchen Modellierung zu bewerten. Somit könnten die vielfach notwendigen, wiederholten Überprüfungen entfallen bzw. deutlich vereinfacht werden.

Um dem Nutzer einen möglichst reibungslosen Übergang zu ermöglichen, sollte der primäre Charakter einer Einzelplatzanwendung erhalten und das zentrale Datenarchiv weitestgehend verborgen bleiben. Des Weiteren erscheint es sinnvoll, für die Aufbereitung der hinterlegten Informationen (Data Mining) eine unabhängige Lösung zu entwickeln. Diese soll primär die Recherche und die Kontrolle der hinterlegten Informationen erleichtern und auch nicht mit dem European Tax Analyzer vertrauten Anwendern einen Zugang ermöglichen. Diese Benutzer dürfte insbesondere die Dokumentation der Steuersysteme interessieren.

---

<sup>7</sup> Vgl. Stetter, T., VITAX, 2005, S. 170.



Deswegen wird nun in der vorliegenden Arbeit das bisherige Datenmodell für eine möglichst versionsunabhängige Speicherung und Archivierung optimiert bzw. neu entwickelt, sodass es die oben genannten Anforderungen erfüllen kann.

### **1.3 Vorgehensweise**

Im zweiten Kapitel wird der European Tax Analyzer detailliert beschrieben. Im Anschluss werden daraus in Kapitel drei die Anforderungen an die persistente, mehrbenutzerfähige Speicherung und die Voraussetzungen für eine effiziente Unterstützung der Zeitreihenanalyse abgeleitet. Insbesondere die Dokumentations-, Recherche- und Kontrollfunktion sind zentrale Aspekte. Zur Unterstützung dieser Funktionen wird die Notwendigkeit einer Benutzerverwaltung und Versionshistorie begründet.

In Kapitel vier erfolgt die Evaluation geeigneter Methoden zur Umsetzung dieser Vorgaben. Zunächst werden die bisherigen Konzepte für die Organisation und Speicherung der notwendigen Daten beleuchtet. Aufgrund der Anforderungen an den Mehrbenutzerbetrieb erscheint ein Datenbankmanagementsystem (DBMS) eine sinnvolle Plattform, so dass sich die weiteren Ausführungen mit den konkreten Anforderungen an ein Datenbankschema beschäftigen. Danach werden die Vor- und Nachteile relationaler und objektorientierter Datenbanken erörtert. Zur Unterstützung der Recherche wird die Verwendung einer webbasierten Lösung in Erwägung gezogen. Danach werden noch prinzipielle Überlegungen zur Vorgehensweise und zu Sicherheitsaspekten angestellt.

In Kapitel fünf werden die notwendigen Änderungen und Erweiterungen detailliert beschrieben und das finale Datenbankschema generiert. Zu Beginn werden die Möglichkeiten zur Unterstützung der Transaktionsverwaltung beleuchtet. Danach wird aus der Notwendigkeit der Unterstützung zweier verschiedener Speicherstrategien die notwendige Kopierinfrastruktur abgeleitet und umgesetzt. In der Folge werden alle Änderungen und Erweiterungen am Datenmodell erläutert sowie die Umsetzung der PDF- und Benutzerverwaltung beschrieben. Des Weiteren wird die Versionshistorie zur Qualitätssicherung vorgestellt. Anschließend wird das im Reverse-Engineering-Verfahren<sup>8</sup> gewonnene Schema formal überprüft. Abschließend erfolgt

---

<sup>8</sup> Das Datenbankschema wird aus dem bereits vorliegenden Programmcode abgeleitet.

die Darstellung von Modifikationen, die keinen direkten Einfluss auf das Datenmodell haben.

Kapitel sechs demonstriert anhand der Berechnung einer Zeitreihe von Deutschland die Funktionen des entwickelten Systems. Hierbei werden alle notwendigen Schritte vom Anlegen eines Benutzers, der Eingabe der ökonomischen Parameter und der Steuersysteme bis hin zu Berechnung von alternativen Szenarien erläutert. Ebenso werden die Möglichkeiten des implementierten Webinterface demonstriert. Abschließend wird die Sicherung der Datenbank beleuchtet.

Das siebte Kapitel fasst die Ergebnisse zusammen.

## 2 European Tax Analyzer

### 2.1 Überblick

Grundlage des Modells ist ein detailliertes Unternehmensmodell, das die ökonomische Entwicklung einer Kapitalgesellschaft i. d. R. über 10 Perioden beschreibt. Dabei werden sämtliche entscheidungsrelevanten Besteuerungskonzeptionen, Steuersysteme und -arten, Tarife sowie Vorschriften zur Ermittlung der Bemessungsgrundlage berücksichtigt. Darüber hinaus kann auch die Ebene der Anteilseigner betrachtet werden.

Prinzipiell ist es recht unwahrscheinlich, dass ein Steuersystem über einen längeren Zeitraum hinweg konstant bleibt. Somit stellen 10 Zeitabschnitte<sup>9</sup> einen guten Kompromiss dar, um auf der einen Seite intertemporale Effekte zu beleuchten und auf der anderen Seite der problematischen Annahme eines konstanten Steuersystems kein zu großes Gewicht zu verleihen.

Die Simulation geht von einem unvollkommenen Kapitalmarkt aus, sodass der Zeitpunkt des Anfalls von konkreten Steuerzahlungen sich durch unterschiedlich hohe Haben- und Sollzinsen direkt auf die Liquidität und Verschuldung des Unternehmens auswirkt und sich dadurch im Unternehmenswert niederschlägt. Die Differenz der Zinssätze verstärkt diese temporalen Effekte. Ein gutes Beispiel hierfür stellen die Abschreibungen von Wirtschaftsgütern dar. Diese können in der Regel nur bis maximal 100 % der Anschaffungs- oder Herstellungskosten (AHK) abgeschrieben werden. Wird der zu versteuernde Gewinn durch vorteilhafte Abschreibungsbedingungen bereits zu Beginn stark gemindert, verschieben sich Steuerzahlungen in die Zukunft und im Unternehmen verbleibt mehr Liquidität. Dadurch kann eine evtl. notwendige Kreditaufnahme verhindert oder gemindert und evtl. sogar Zinseinnahmen erzielt werden. Daher macht es einen erheblichen Unterschied, ob der größte Teil der gesamten Abschreibungen schon während der ersten Jahre der Nutzungsdauer den zu versteuernden Gewinn mindert oder erst später.

Teil des Modells ist ein detaillierter Produktions- und Absatzplan, der genau festlegt, wie viele Produkte produziert und abgesetzt werden. Auch sämtliche Kostenbestand-

---

<sup>9</sup> Vgl. Spengel, C., Steuerbelastungsvergleiche, 1995, S. 161.

teile, wie z. B. Personal und Material, müssen eingegeben werden. Des Weiteren sind alle Gebäude, Maschinen, Patente und Lizenzen, Büro- und Geschäftsausstattungen usw. zu hinterlegen. Die genaue Finanzierungsstruktur (wie z. B. Gesellschafterdarlehn), Informationen bzgl. Gewährleistungen und Energieverbrauch, sowie detaillierte Daten zu weiteren Einnahmen und Ausgaben sind im System zu erfassen.<sup>10</sup>

Aufgrund der Komplexität der benötigten Daten ist es nicht möglich, diese aus Unternehmensdatenbanken wie z. B. *AMADEUS* oder *DAFNE*<sup>11</sup> zu extrahieren. Daher werden in der Regel auf Basis dieser Datenbanken verschiedene Unternehmenskennzahlen ermittelt, die als Referenz für das Vergleichsunternehmen dienen.

Die bisher aufgelisteten Angaben stellen primär rein ökonomische Sachverhalte dar und haben i. d. R. nur indirekte Auswirkungen auf die Ermittlung der Steuerbelastung. Diese Daten, bzw. aus diesen abgeleitete Zwischenergebnisse, stellen häufig die Anknüpfungspunkte für die Bemessungsgrundlagen einzelner Steuern dar. Wenn das Archiv für Steuerinformationen mehr als nur Tarife einzelner Steuern enthalten soll, ist zumindest eine Möglichkeit der Rekonstruktion solcher Verknüpfungen notwendig.

Des Weiteren werden Regeln zur Ermittlung von Zeitwerten (z. B. Abschreibungsvorschriften zur Ermittlung von Restbuchwerten), zur Bewertung von Lagerbeständen (Verbrauchsfolge und Herstellungskostenermittlung) und Pensionsverpflichtungen sowie zur Ausübung von weiteren Wahlrechten benötigt. Im Modell werden diese Optionen immer steueroptimal ausgeübt, sodass nicht alle erlaubten Varianten notwendig sind.

Zu guter Letzt werden auch noch die genauen Bestimmungen zur Ermittlung der Bemessungsgrundlagen und die Tarife der einzelnen Steuern benötigt.

Für die Anteilseignerebene werden Angaben bzgl. der Beteiligungshöhe, des sonstigen Vermögens, der Arbeitseinkünfte und des Anteils am Gesellschafterdarlehn benötigt, um nur einige zu nennen. Eine Besonderheit stellt das virtuelle Einkommen dar, das an sich keine ökonomischen Effekte verursacht, aber das zu versteuernde Einkommen erhöht, um Progressionseffekte bei der Einkommensteuer besser abbil-

---

<sup>10</sup> Für eine detailliertere Darstellung siehe Spengel, C., Steuerbelastungsvergleiche, 1995.

<sup>11</sup> *AMADEUS* und *DAFNE* sind Unternehmensdatenbanken vom Bureau van Dijk (BvD).

den zu können. Dies ist notwendig, um Einkünfte außerhalb des Modellrahmens erfassen zu können, ohne die Vergleichbarkeit der Ergebnisse zu beeinflussen. Diese Angaben haben keinen direkten Einfluss auf die Steuerberechnung. Somit bleiben nur die konkreten Regelungen bezüglich der Bemessungsgrundlagen und Tarife übrig und die Anknüpfung der Bemessungsgrundlagen an diese Einkunftsarten und Vermögenswerte. Bilanzielle Wahlrechte,<sup>12</sup> wie auf Unternehmensebene, gibt es nicht.

Der European Tax Analyzer errechnet aus diesen Informationen einen Unternehmensendwert nach 10 Perioden für den Fall mit und ohne Steuern. Die Differenz der beiden Werte ergibt die Steuerbelastung des Unternehmens. Diese weicht im Normalfall deutlich von der reinen Summe der Steuerzahlungen ab. Neben dem Zinseszinseffekt spiegelt dieser Wert auch den temporalen Anfall und dessen Auswirkungen auf die Liquidität wider. So kann z. B. die Steuerbelastung auch dadurch steigen, dass das Unternehmen die Steuerzahlung zwischenfinanzieren musste und keine überschüssige Liquidität zum kurzfristigen Habenzins anlegen konnte.

Dieser Vergleich kann auf der Unternehmensebene und auf der Gesamtebene (Unternehmen plus Anteilseigner) erfolgen. Bei der isolierten Betrachtung ohne Gesellschafter ist jedoch auf gleichbleibende Ausschüttungen zu achten, die nicht von steuerlichen Größen abhängig sind, da diese den Wert der Firma mindern und anderenfalls zu Verzerrungen führen würden.

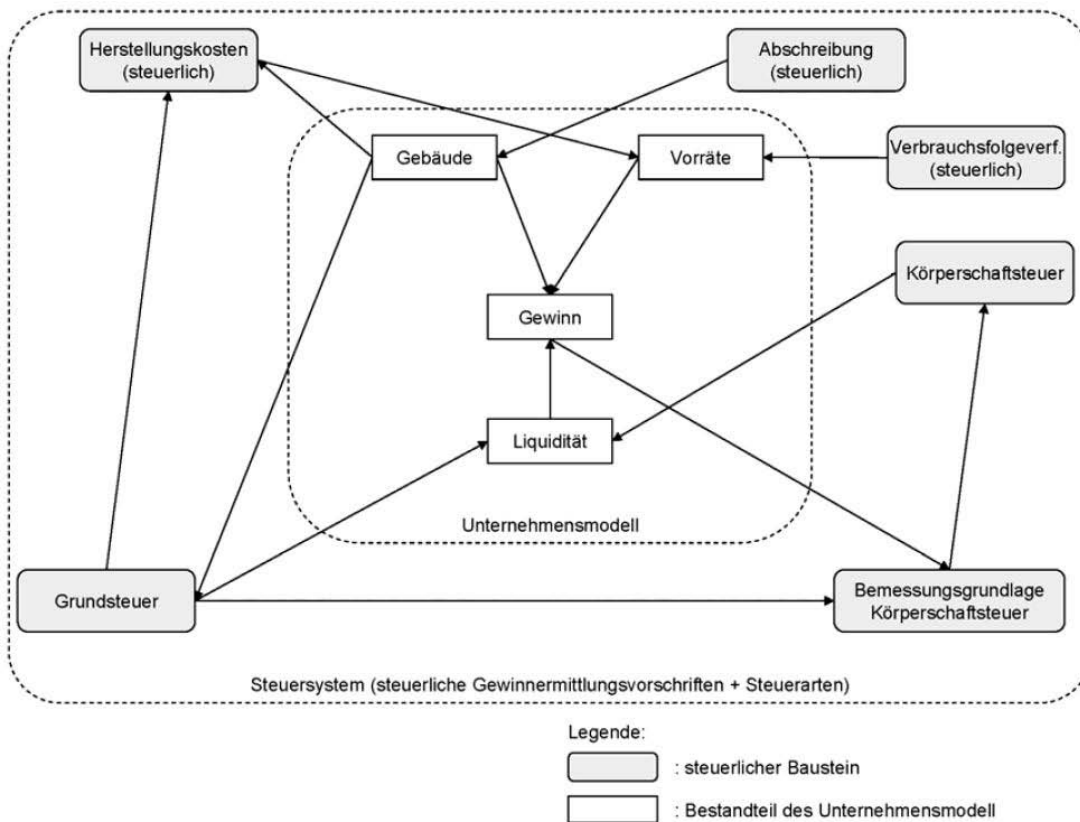
Die am häufigsten vorkommenden Steuerarten sind auf Unternehmensebene die Grundsteuern, Körperschaftsteuern, Gewerbesteuern, lokale Substanzsteuern sowie auf der Anteilseignerebene die Einkommensteuer. Diese beschränkt sich jedoch meistens auf Kapitaleinkünfte (Dividenden, Zinsen und Veräußerungsgewinne).

---

<sup>12</sup> Das bei Personengesellschaften übliche Sonderbetriebsvermögen wird nur indirekt abgebildet. Bei der Vermietung & Verpachtung verbleibt z. B. das Gebäude im Unternehmen und Miete und/oder Abschreibung werden an den Gesellschafter weitergeleitet. Bei der Modellierung der Steuern kann auf das Gebäude entsprechend zurückgegriffen werden.

## 2.2 Beschreibung

Abbildung 1: Interaktion zwischen Unternehmensmodell und Steuern



Quelle: Stetter, T., ViTax, 2005, S. 63.

Abbildung 1 demonstriert beispielhaft die wechselseitige Beziehung zwischen der Ermittlung der Steuern und dem Fortschreiben des Unternehmensmodells. Zur Ermittlung der Grundsteuer sind etwa Informationen über die Gebäude der Gesellschaft notwendig. Zugleich kann sie auch bei den Herstellungskosten angesetzt werden. Dabei beeinflusst sie über die Bewertung der Vorräte in Verbindung mit der gewählten Verbrauchsfolge den ausgewiesenen Gewinn. Der gleiche Effekt gilt auch für die Abschreibungen der entsprechenden Immobilien. Die Grundsteuer kann zudem von der Bemessungsgrundlage der Körperschaftsteuer abgezogen werden. Beide Steuern beeinflussen direkt die Liquidität des Unternehmens und damit das Finanzergebnis der nächsten Periode. Das Beispiel bildet nur einen kleinen Ausschnitt möglicher Interaktionen ab, da die Komplexität des Modells deutlich höher ist.

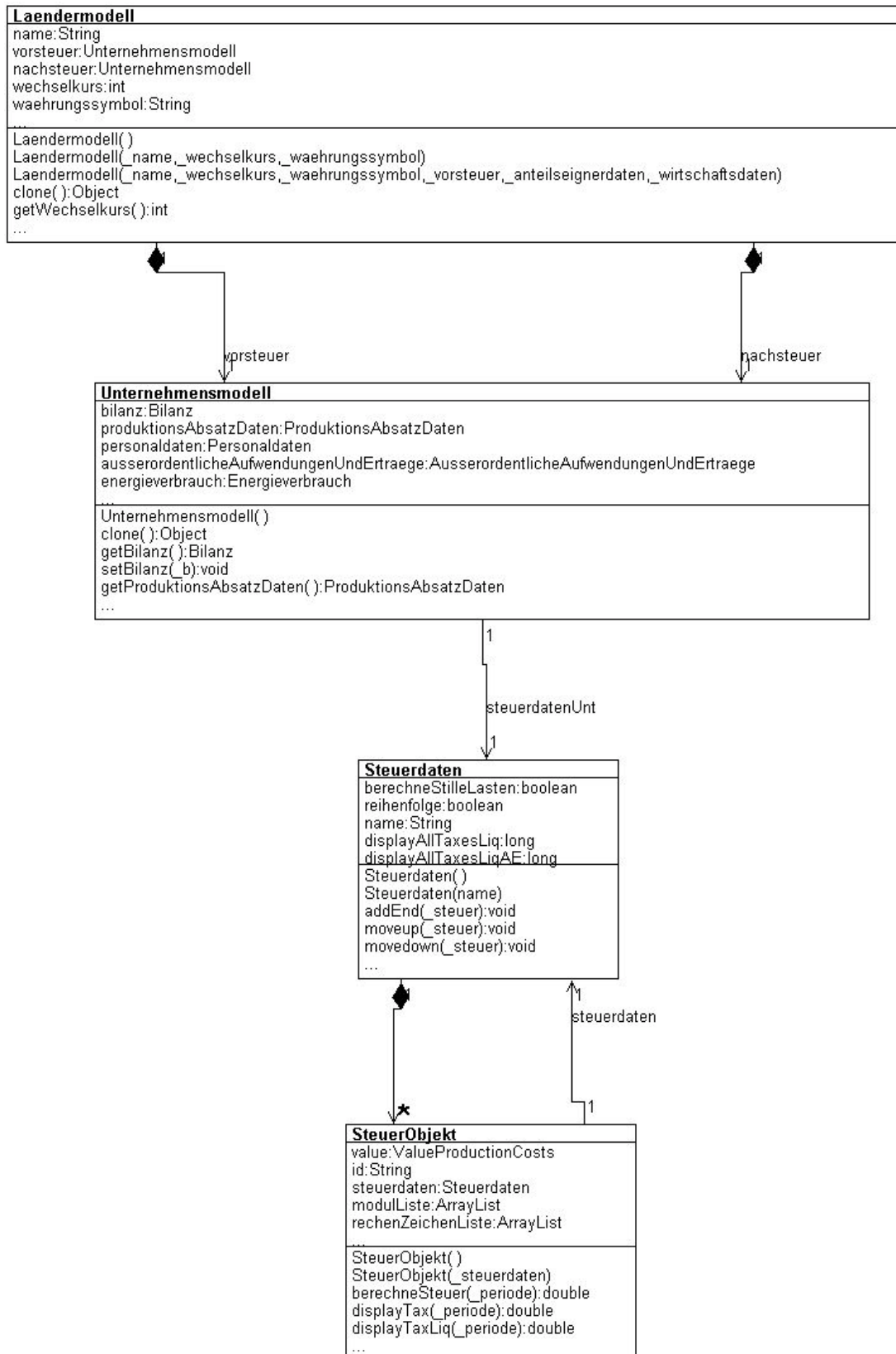
### 2.2.1 Benötigte Informationen und bisheriges Datenmodell

Als Grundlage dient ein Unternehmen aus dem verarbeitenden Gewerbe. Die Auswahl hat damit Einfluss auf die Zusammensetzung der Wirtschaftsgüter (z. B. werden 5 Maschinen abgebildet) und den Produktions- und Absatzplan. Dieses Beispiel

stellt jedoch nur den modelltheoretischen Rahmen dar und hat keinerlei Auswirkung auf die Möglichkeit der Modellierung anderer Branchen. Im Transportgewerbe könnten die Maschinen z. B. als Teil der Lastkraftwagenflotte und die produzierten Güter als transportierte interpretiert werden. Zudem wird eine Kapitalgesellschaft unterstellt, so dass die Anteilseignerebene und die Unternehmensebene getrennt modelliert und berechnet werden. Zur Unterscheidung der Steuersysteme einzelner Staaten werden diese Informationen in Länderobjekten gekapselt.

Ein Land besteht (vgl. Abbildung 2) aus jeweils zwei Instanzen des Unternehmensmodells und den Anteilseignerdaten, den wirtschaftlichen Rahmendaten (Inflations- und Preissteigerungsraten), dem Wechselkurs, dem Währungssymbol und dem eigentlichen Namen. Für die Benennung der Länder gibt es zurzeit keine einheitlichen Regelungen, sodass i. d. R. Jahr und Art der Modellierung in der Namensgebung berücksichtigt werden müssen.

Abbildung 2: Ländermodell, Unternehmensmodell und Steuerdaten



Quelle: Stetter, T., ViTax, 2005, S. 141.



Das Unternehmensmodell beinhaltet alle restlichen Informationen. Nur die zur Durchführung eines Berechnungslaufes nicht länderspezifischen Parameter werden separat behandelt. Es setzt sich aus Daten über:

- die Bilanz,
- die Produktions- und Absatzdaten,
- die Personaldaten,
- die außerordentlichen Aufwendungen und Erträge,
- die Aufwendungen für Forschung und Entwicklung,
- die Materialkosten,
- die sonstige Einnahmen und Ausgaben,
- die Gewährleistungen,
- den Lagerbestand,
- das Geschäftsführergehalt,
- die Herstellungskosten,
- den Energieverbrauch,
- das verwendbare Eigenkapital,
- das Betriebsergebnis,
- das Finanzergebnis,
- das Einkommen,
- die Liquidität,
- und das Steuersystem zusammen.

Im Fall ohne Steuern ist die letzte Referenz ungültig („null“).

Tabelle 1 zeigt die einzelnen Positionen der Bilanz. Hierbei geben die Zahlen in Klammern die Anzahl der Wirtschaftsgüter an. Es gibt z. B. ein Fabrik- und ein Bürogebäude.

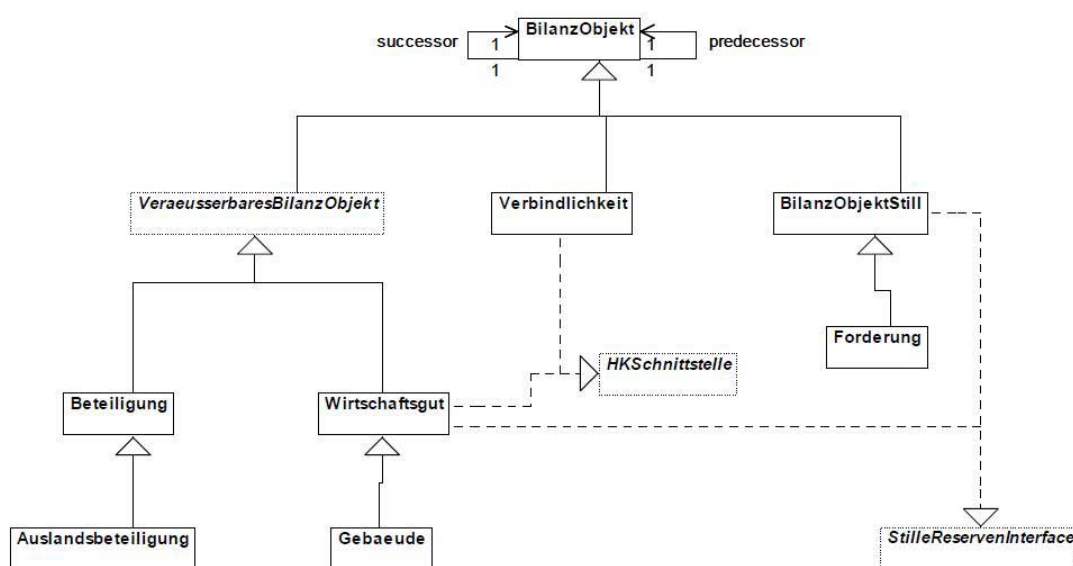
Tabelle 1: Vermögens- und Kapitalausstattung (Bilanz)

AKTIVA		PASSIVA	
<b>A. Anlagevermögen</b>		<b>A. Eigenkapital</b>	
<i>I. Immaterielle Vermögensgegenstände</i>		<i>I. Stammkapital</i>	
1. Patente	(1)	<i>B. Rückstellungen</i>	
2. Lizenzen	(1)	1. Rückstellungen für Pensionen	
<i>II. Sachanlagen</i>		2. Sonstige Rückstellungen	
1. Unbebaute Grundstücke	(1)	<i>C. Verbindlichkeiten</i>	
2. Bebaute Grundstücke	(2)	<i>I. Verbindlichkeiten gegenüber Dritten</i>	
3. Maschinen	(5)	<i>II. Verbindlichkeiten gegenüber Anteilseignern</i>	
4. Betriebs- und Geschäftsausstattung	(1)	<i>III. Verbindlichkeiten aus Lieferungen und Leistungen</i>	
<i>III. Finanzanlagen</i>		<i>IV. Kurzfristige Verbindlichkeiten</i>	
1. Beteiligungen	(2)		
2. Forderungen	(3)		
<b>B. Umlaufvermögen</b>			
I. Vorräte			
II. Forderungen aus Lieferungen und Leistungen			
III. Bankguthaben			

Quelle: Vgl. Gutekunst, G., Steuerbelastungen, 2005, S. 67.

Zurzeit sind die Summen der einzelnen Bilanzpositionen einzugeben und anschließend prozentual auf die einzelnen Wirtschaftsgüter zu verteilen. Bei einzelnen Positionen können stille Lasten oder Reserven (*StilleReservenInterface* bzw. *BilanzObjektStill*) auftreten und/oder müssen nach Ablauf der wirtschaftlichen Nutzungsdauer (*Wirtschaftsgut*) ersetzt werden. Des Weiteren können sie evtl. bereits vorher veräußert (*VeraeusserbaresBilanzObjekt*) werden. Diese unterschiedlichen Eigenschaften werden durch die Vererbungshierarchie (vgl. Abbildung 3) abgebildet.

Abbildung 3: Vererbungshierarchie der Bilanzobjekte



Quelle: Stetter, T., ViTax, 2005, S. 119.

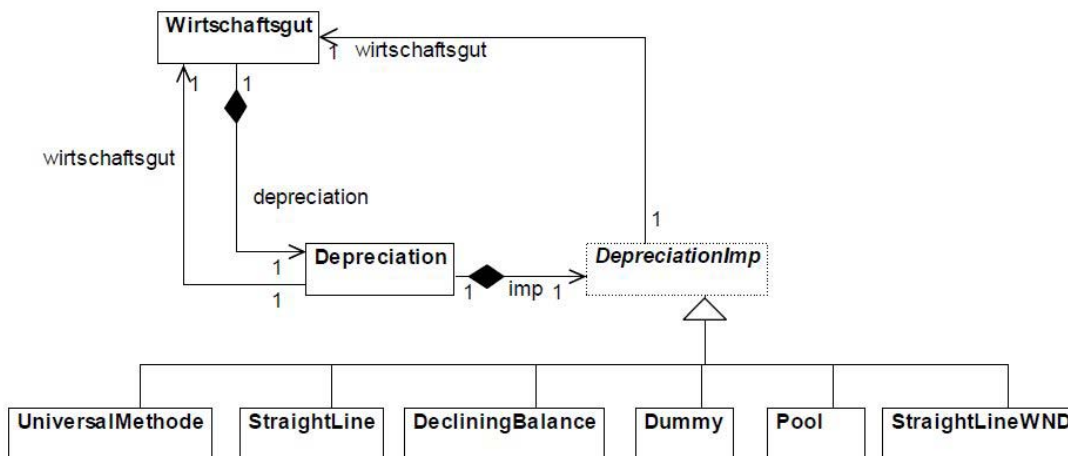
Bei Beteiligungen sind z. B. noch Angaben über die Dividendenhöhe oder den ausländischen Körperschaftsteuersatz notwendig. Für alle Positionen, die in die Herstellungskosten (vgl. *HK Schnittstelle*) einbezogen werden können, muss der Aufwand (z. B. Abschreibungen) auf die drei funktionalen Bereiche Produktion, Verwaltung und Vertrieb aufgeteilt werden.

Um den Werteverzehr des Anlagevermögens abzubilden, müssen Regelungen zur Ermittlung spezifiziert werden. Im einfachsten Fall werden die Anschaffungs- oder Herstellungskosten gleichmäßig auf die wirtschaftliche Nutzungsdauer aufgeteilt. Für eine Maschine, die z. B. 5 Jahre lang genutzt werden kann und zu 100 angeschafft wurde, würden somit 20 als Abschreibungsrate pro Jahr angesetzt. Dieses Wirtschaftsgut besitzt dann einen Restbuchwert von 80 nach einem Jahr. Dieser wird in der Bilanz ausgewiesen. Im zweiten Jahr beträgt dieser Wert dann 60, im dritten 40, im vierten 20 und im fünften ist die Maschine komplett abgeschrieben. Sie wird aus der Bilanz entfernt und wird gleichwertig ersetzt.

Bei der Pool-Methode wird ein fester Prozentsatz vom Restbuchwert abgeschrieben. Im zweiten Jahr beträgt dieser z. B. 16 (20 % von 80). Bei diesem Verfahren wird somit nie ein Wert von Null erreicht, sodass das Wirtschaftsgut nach dem Ausscheiden aus dem Betrieb weiterhin Teil der Bilanz ist und Abschreibungen weiter berechnet werden müssen.

Nicht alle möglichen Abschreibungsverläufe können jedoch über vordefinierte Methoden erfolgen, sodass es eine Universalmethode gibt. Bei dieser kann für jede Periode genau festgelegt werden, welcher absolute oder relative Wert von den Anschaffungskosten oder dem Restbuchwert abzuschreiben ist (vgl. Abbildung 21). Die konkreten Methoden dienen der vereinfachten Eingabe. *Dummy* schaltet die Berechnung aus. Dies ist zum Beispiel beim unbebauten Grundstück gewünscht, da dieses normalerweise keinem Werteverzehr unterliegt.

Abbildung 4: Implementierung der Abschreibungen



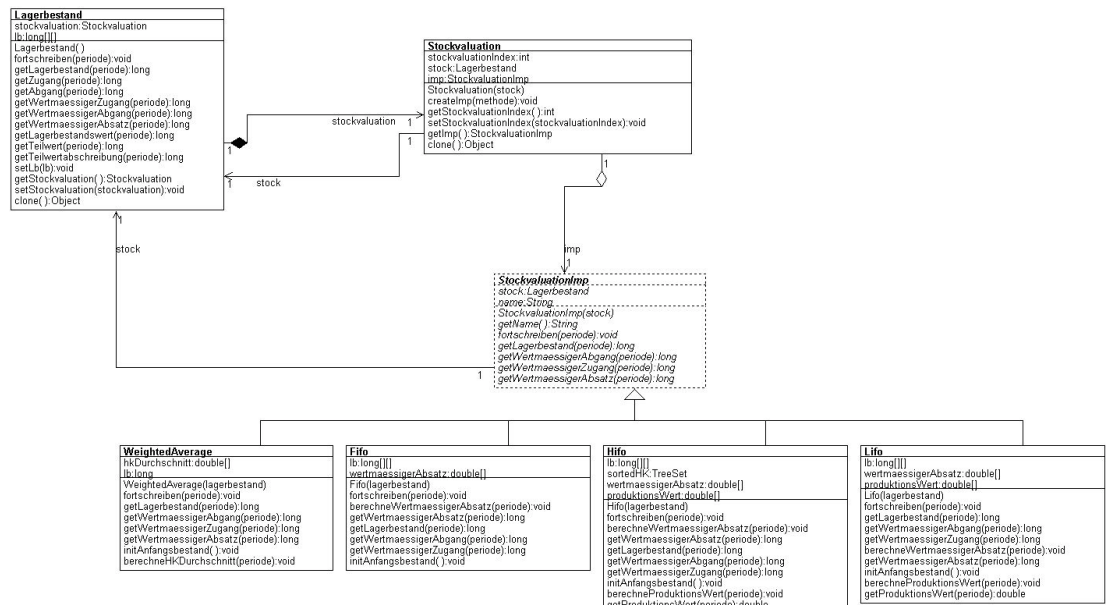
Quelle: Stetter, T., ViTax, 2005, S. 131.

Somit ergibt sich die in Abbildung 4 gezeigte Vererbungshierarchie der einzelnen Abschreibungen und deren Beziehungen zum Wirtschaftsgut.

Ein wichtiges Element der Gewinnermittlung stellt die Vorratsbewertung dar. Aufwendungen für nicht abgesetzte Güter mindern das Ergebnis nicht, da diese in der Bilanz aktiviert und somit gegengebucht werden. Damit nicht jedes einzelne Produkt getrennt zu bewerten ist, gibt es Verfahren zur Vereinfachung, die eine gewisse Reihenfolge des Verbrauchs bzw. der Entnahme unterstellen.

FIFO (First In First Out) unterstellt z. B., dass zuerst das als erstes eingelagerte Produkt entnommen wird. LIFO (Last In First Out) bedeutet, dass der letzte Zugang als erstes verbraucht wird. Bei HIFO (Highest In First Out) wird das teuerste Gut zuerst entnommen. Beim Durchschnittskostenverfahren wird ein durchschnittlicher Preis für alle Produkte ermittelt, sodass die konkrete Reihenfolge des Verbrauchs nicht mehr wichtig ist. Abbildung 5 zeigt die aktuelle Implementierung dieser Methoden.

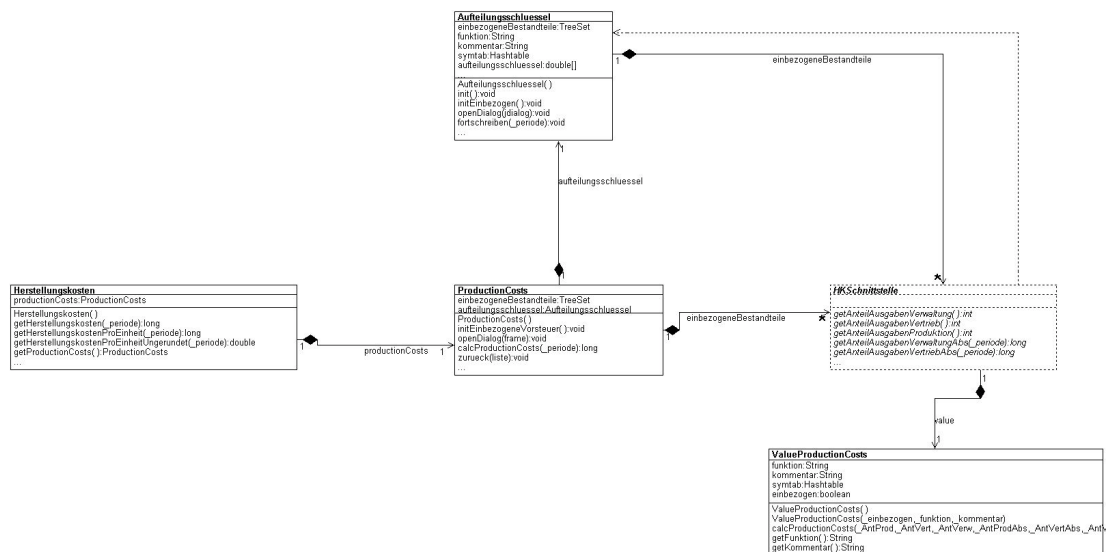
**Abbildung 5: Lagerbestandsverwaltung und Verbrauchsfolge**



*Quelle: Stetter, T., ViTax, 2005, S. 139.*

Um die Herstellungskosten ermitteln zu können, ist es zunächst notwendig zu spezifizieren, welche Kostenart in welchem Umfang in die Berechnung eingeht. Hierfür kann der Benutzer eine Formel (*ValueProductionCosts*) hinterlegen. Variablen bilden dabei die auf die einzelnen Funktionsbereiche (vgl. *HK Schnittstelle*) entfallenden Kosten ab. Somit ist es auf einfache Art und Weise möglich, Teil- oder Vollkosten zu modellieren. Der *Aufteilungsschlüssel* dient der Objektivierung der angemessenen Fertigungsgemeinkosten. Die restlichen Klassen aus Abbildung 6 dienen der Berechnung.

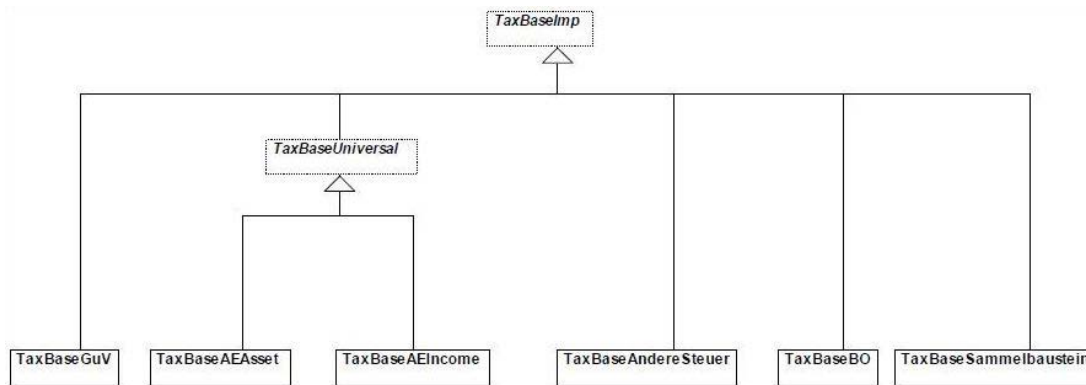
**Abbildung 6: Klassen für die Ermittlung der Herstellungskosten**



*Quelle: Stetter, T., ViTax, 2005, S. 136.*

Die zentrale Innovation der Arbeit von Stetter bestand darin, dem Anwender die Modellierung von Steuern zu ermöglichen. Hierfür wurde eine Steuer (vgl. Abbildung 8) in Tarif (vgl. Abbildung 9) und Bemessungsgrundlage (vgl. Abbildung 7) unterteilt.

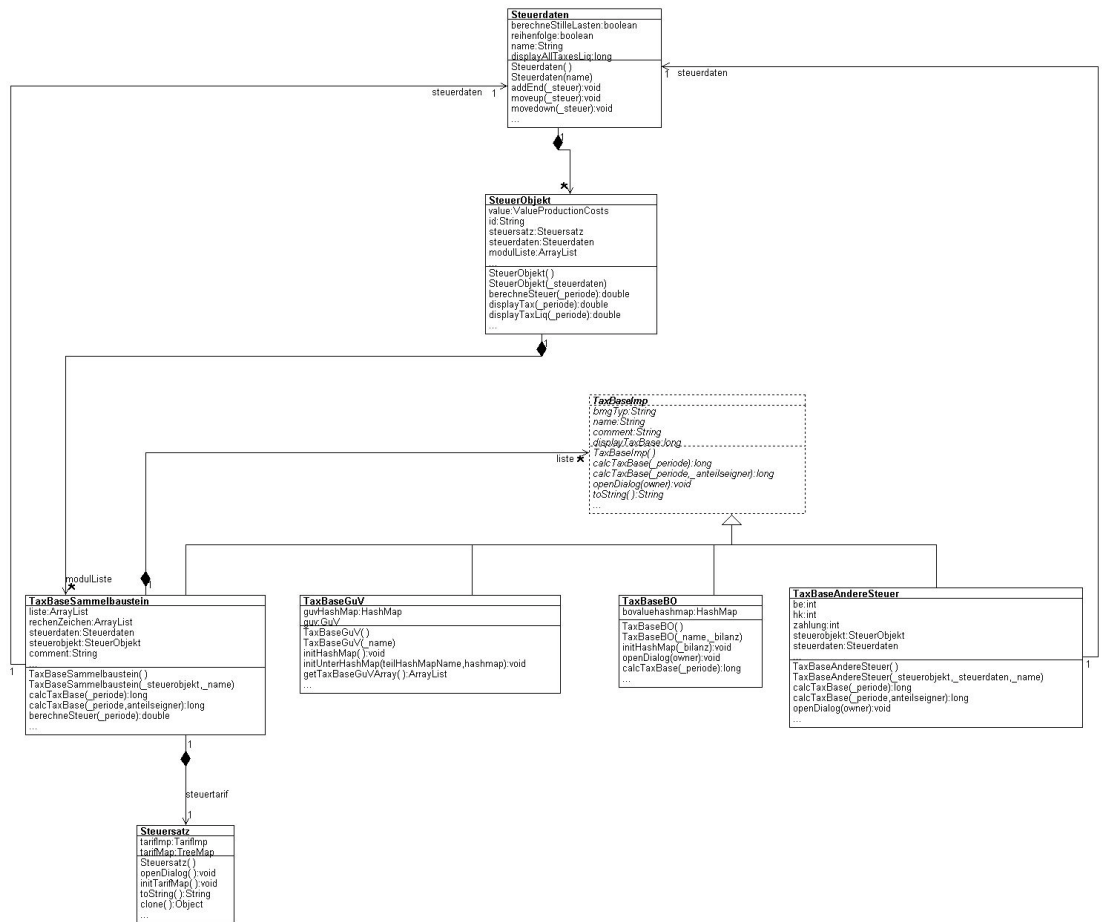
**Abbildung 7: Vererbungshierarchie der Bemessungsgrundlagen**



*Quelle: Stetter, T., ViTax, 2005, S. 150.*

Der *Sammelbaustein* dient als Gliederungselement und enthält neben einer Liste anderer Bemessungsgrundlagen und deren entsprechenden mathematischen Operationen auch den Tarif. Dieser muss aber nicht zwingend den Steuertarif abbilden, sondern kann auch für Zwischenrechnungen benutzt werden. Der *Sammelbaustein* kann sich selbst enthalten, wodurch eine beliebig tiefe Schachtelung (vergleichbar mit Matryoshka-Puppen) möglich ist.

Abbildung 8: Steuerdaten, Steuerobjekte und Bemessungsgrundlagen



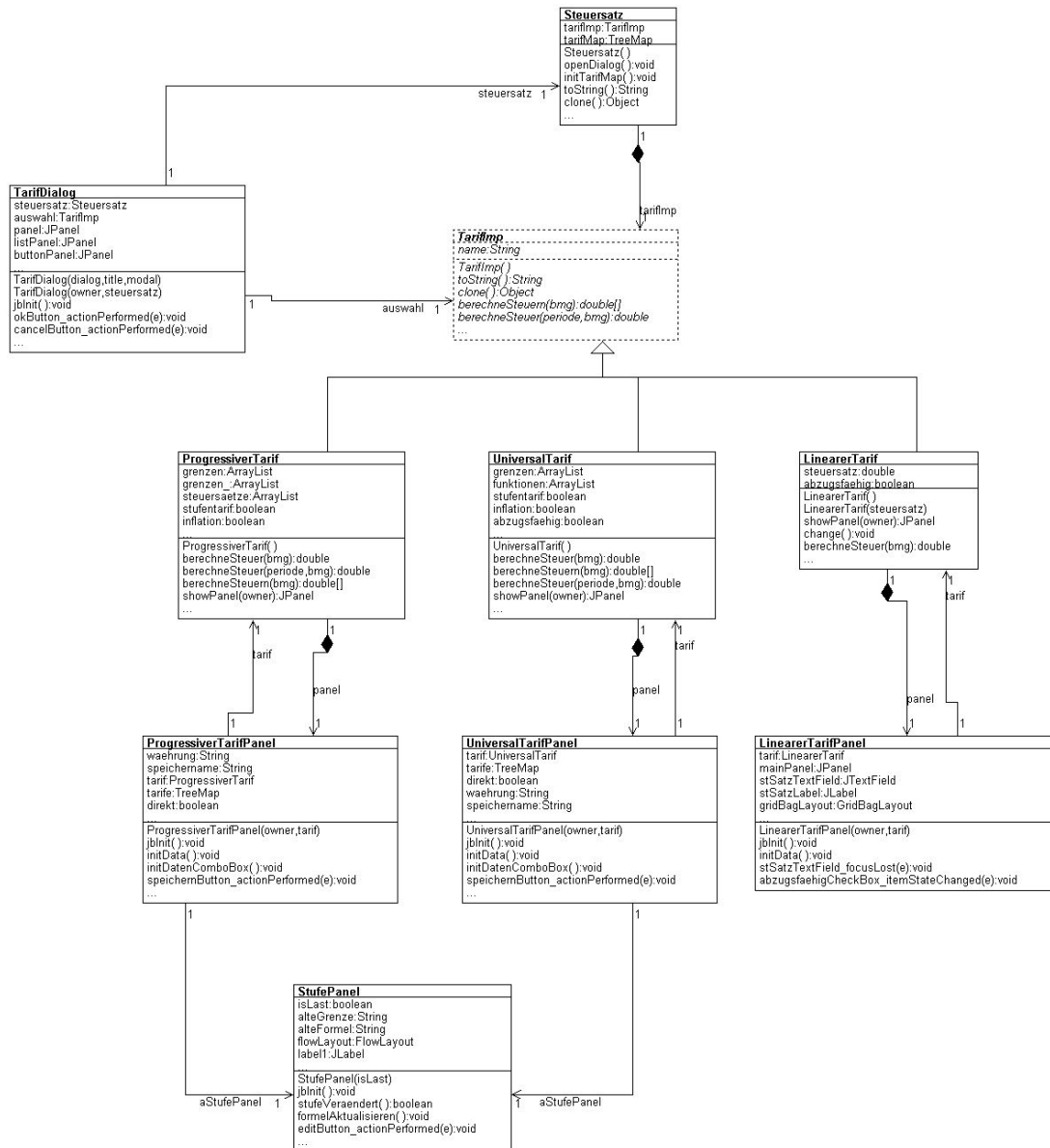
Quelle: Stetter, T., ViTax, 2005, S. 147.

Die *Steuerdaten* (vgl. Abbildung 8) bilden ein Steuersystem ab, das beliebig viele Steuern (*SteuerObjekt*) enthalten kann. Dies enthält eine Liste (*modulListe*) von *Sammelbausteinen*.

Zurzeit gibt es drei mögliche Tarife (vgl. Abbildung 9). Der lineare Tarif multipliziert das Resultat der Liste der Bemessungsgrundlagen aus dem Sammelbaustein mit einem prozentualen Wert. Der progressive Tarif kann zusätzlich verschiedene Stufen definieren, für die jeweils separate Steuersätze hinterlegt werden können. Der Universaltarif ermöglicht die Spezifikation beliebiger Formeln anstelle prozentualer Steuersätze.

Für die Anteilseignerebene erfolgt die Modellierung der einzelnen Steuersysteme identisch. Allerdings können hier mehrere Systeme hinterlegt werden und verschiedenen Anteilseignern zugewiesen werden. So könnte z. B. ein Anteilseigner verheiratet sein und daher den Splittingtarif nutzen können, während ein anderer dies nicht kann.

Abbildung 9: Implementierung der Steuertarife



Quelle: Stetter, T., ViTax, 2005, S. 163.

### 2.2.2 Berechnung der Steuerbelastung

Nachdem die Parameter für das Unternehmensmodell spezifiziert und die Steuersysteme eingegeben worden sind, kann die eigentliche Berechnung stattfinden. Innerhalb des zehnjährigen Betrachtungshorizontes wird von gleichbleibenden wirtschaftlichen und steuerlichen Rahmenbedingungen ausgegangen. Wiederbeschaffungskosten eines Wirtschaftsgutes werden z. B. durch die Inflation der Anschaffungskosten



mit der Preissteigerungsrate für Investitionsgüter ermittelt. Ebenso steigen die Löhne und Gehälter jede Periode um die Rate der Tarifabschlüsse an.<sup>13</sup>

Das Unternehmensmodell wird einmal mit und ohne Steuersystem berechnet. Aus der aus dem Endvermögensvergleich<sup>14</sup> berechneten Differenz resultiert dann die Steuerbelastung. Hierbei ist jeder Vermögensgegenstand, jede Forderung sowie alle Verbindlichkeiten und Verpflichtungen einzeln zu bewerten.<sup>15</sup> Diese normierten Wertansätze berücksichtigen jedoch keinerlei stille Reserven und Lasten.<sup>16</sup> Diese treten auf, wenn z. B. der steuerliche Wertansatz vom Ergebnis des Bewertungsverfahrens abweicht. Eine stille Reserve entsteht z. B. bei einer Maschine, wenn ihr steuerlicher Wert niedriger ist. Eine stille Last entsteht z. B., wenn die tatsächlichen Pensionsverpflichtungen höher sind, als diese in der Steuerbilanz ausgewiesen sind. Wären diese Verpflichtungen niedriger, ergebe sich auch eine Reserve.

Das Unternehmen ist auf Dauer angelegt, sodass sich aus den Annahmen zur Unternehmensentwicklung am Ende des Planungshorizontes keine Informationen über die Auflösung dieser stillen Reserven und Lasten ergeben.<sup>17</sup> Dies ist jedoch ein wichtiger Aspekt, da der Abbau dieser Bewertungsdifferenzen auch steuerwirksam ist und die Höhe der Steuerzahlungen in Zukunft verändern wird. Diese Änderungen wurden jedoch schon während des Planungshorizontes verursacht und sollten daher auch das berechnete Endvermögen beeinflussen. Somit sind Annahmen zu treffen wie diese behandelt werden. Im Normalfall werden daher alle stillen Lasten und Reserven am Ende des Planungshorizontes gehoben und dem steuerlichen Gewinn der letzten Periode hinzugerechnet.

In Tabelle 2 wird am Beispiel eines in Periode eins angeschafften Wirtschaftsgutes die Entstehung von stillen Lasten und Reserven für zwei unterschiedliche Länder demonstriert und der daraus resultierende Steuervorteil für einen Belastungsvergleich aufgezeigt.<sup>18</sup> Hierbei wird von Anschaffungskosten in Höhe von 100.000 € und einer

---

<sup>13</sup> Für einen vollständigen Überblick siehe Meyer, R., Simulation von Steuerbelastungen, 1996, S. 136 ff.

<sup>14</sup> Vgl. Spengel, C., Steuerbelastungsvergleiche, 1995, S. 197 f.

<sup>15</sup> Für detaillierte Bewertungsverfahren einzelner Sachverhalte vgl. Spengel, C., Steuerbelastungsvergleiche, 1995, S. 197 ff.

<sup>16</sup> Für einen vollständigen Überblick siehe Gutekunst, G., Steuerbelastungen, 2005, S. 96-101.

<sup>17</sup> Spengel, C., Steuerbelastungsvergleiche, 1995, S. 200.

<sup>18</sup> Vgl. Gutekunst, G., Steuerbelastungen, 2005, S. 98-99.

wirtschaftlichen Nutzungsdauer von 4 Jahren ausgegangen. Beide Länder haben jeweils einen Steuersatz von 40%.

Tabelle 2: Beispiel zur Entstehung von stillen Lasten und Reserven (Angaben in €)

	Periode			
	1	2	3	4
<b>Vorsteuerfall</b>				
Restbuchwert 1.1.	1.000.000	750.000	500.000	250.000
Abschreibung	250.000	250.000	250.000	250.000
Restbuchwert 31.12.	750.000	500.000	250.000	0
<b>Nachsteuerfall Land 1</b>				
Restbuchwert 1.1.	1.000.000	600.000	360.000	180.000
Abschreibung	400.000	240.000	180.000	180.000
Restbuchwert 31.12.	600.000	360.000	180.000	0
<b>Nachsteuerfall Land 2</b>				
Restbuchwert 1.1.	1.000.000	875.000	625.000	375.000
Abschreibung	125.000	250.000	250.000	375.000
Restbuchwert 31.12.	875.000	625.000	375.000	0
<b>Steuerliche Konsequenz</b>				
Bemessungsgrundlagendifferenz Land 1 (Wirtschaftsgut)	- 150.000	+ 10.000	+ 70.000	+ 70.000
Bemessungsgrundlagendifferenz Land 2 (Wirtschaftsgut)	+ 125.000	0	0	- 125.000
Steuervorteil Land 1 ggü. Land 2 (Ertragsteuersatz 40%)	+ 110.000	- 4.000	- 28.000	- 78.000
<b>Stille Reserve Land 1</b>		<b>140.000</b>		
<b>Stille Last Land 2</b>		<b>125.000</b>		

Quelle: Gutekunst, G., Steuerbelastungen, 2005, S. 98.

Im ersten Land kann degressiv mit Übergang zur linearen Methode<sup>19</sup> in Höhe von 40% abgeschrieben werden. Dagegen kann im zweiten Land nur linear abgeschrieben und im ersten Jahr nur die halbe Jahresabschreibung angesetzt werden. Im Vorsteuerfall wird linear abgeschrieben. „Zwar gleichen sich die Bemessungsgrundlagendifferenzen über die gesamte ökonomische Nutzungsdauer des Wirtschaftsgutes in den einzelnen Ländern sowie im Vergleich zum Vorsteuerfall vollständig aus, gleichwohl ergeben sich dann erhebliche Differenzen, sofern der Planungshorizont begrenzt ist und die ökonomische Nutzungsdauer der Wirtschaftsgüter unterschreitet.“<sup>20</sup> Im konkreten Beispiel wird von einem Planungshorizont von zwei Perioden

<sup>19</sup> Vgl. Ihrig, H./Pflaumer, P., 2009, S. 149 f.

<sup>20</sup> Gutekunst, G., Steuerbelastungen, 2005, S. 98.

ausgegangen, sodass zu diesem Zeitpunkt Bewertungsunterschiede auftreten, die die Steuerbelastung verzerren und somit den Endvermögensvergleich beeinflussen. Im ersten Land entsteht eine stille Reserve, da der Restbuchwert geringer ist als im Vorsteuerfall. Die erhöhte Abschreibung in der ersten Periode hat den steuerlichen Gewinn stärker gemindert als dies bei der linearen Methode der Fall gewesen wäre und führt somit zu einem temporären Steuervorteil. Dieser Vorteil wird durch geringere Abschreibungen in den folgenden Perioden sukzessive wieder ausgeglichen.

In Periode 2 beträgt der „gegenwärtige Steuervorteil von Land 1 gegenüber Land 2 unter Vernachlässigung von steuerpflichtigen Zinseffekten 106.000 € [110.000 € – 4.000 €].“<sup>21</sup> Durch den Einbezug der stillen Lasten und Reserven in die Bewertungskonzeption können die Unterschiede, die aus latenten Steuerfolgen<sup>22</sup> resultieren, nahezu ausgeglichen werden.<sup>23</sup>

Nicht genutzte Verlustvorträge stellen insofern ein ähnliches Problem dar, da diese die zukünftigen Steuerzahlungen mindern könnten.<sup>24</sup> Diese entstehen, wenn das Unternehmen Verluste nicht mit Gewinnen aus den Folgeperioden<sup>25</sup> verrechnen konnte. Eine direkte Steuererstattung gibt es i. d. R. nicht.

Bei der Bewertung der Verlustvorträge besteht eine zusätzliche Unsicherheit, da je nach konkreter steuerlicher Ausgestaltung die zukünftige Nutzung evtl. ganz entfallen kann. Häufig verfallen Verlustvorträge z. B. bereits 5 Jahre nach deren Entstehung. Konnte das Unternehmen während dieser Zeit keine steuerlichen Gewinne erzielen, verfällt der Vortrag komplett und hätte in diesem Fall keinerlei Wert. Im Falle der zukünftigen Nutzung mindern die Verluste die steuerlichen Gewinne und reduzieren somit die Steuerzahlung. Der resultierende Wert wäre in diesem Fall die abge-

---

<sup>21</sup> Gutekunst, G., Steuerbelastungen, 2005, S. 99.

<sup>22</sup> Aktuelle Sachverhalte können Auswirkungen auf zukünftige Steuerzahlungen haben. Diese Folgen sind häufig nicht auf den ersten Blick ersichtlich und die Höhe und der Zeitpunkt des Eintretens ungewiss.

<sup>23</sup> Im Gegensatz zur Arbeit von Gutekunst (vgl. Gutekunst, G., Steuerbelastungen, 2005, S. 100 f.) werden zurzeit nur absolute und keine gewogenen Restbuchwertdifferenzen ermittelt.

<sup>24</sup> Für eine vollständige Darstellung der Problematik siehe Sachverständigenrat zur Begutachtung der Gesamtwirtschaftlichen Entwicklung, et al., Duale Einkommensteuer, 2006, S. 139-146.

<sup>25</sup> Evtl. mögliche Verlustrückträge mindern die verbleibenden Verlustvorträge zusätzlich, jedoch sind diese meistens stark limitiert, wenn sie überhaupt erlaubt sind.

zinste Reduktion der Steuerzahlung.<sup>26</sup> Diese Minderung der zukünftigen Steuerzahlungen wurde jedoch bereits während des Planungshorizontes verursacht und muss ähnlich wie bei den stillen Lasten und Reserven bei der Bewertungskonzeption berücksichtigt werden. Aufgrund der erhöhten Unsicherheit bezüglich der zukünftigen Nutzung, werden auf Basis der steuerlichen Restriktionen Bewertungsabschläge ermittelt. Im Normalfall werden dann die bewerteten Verlustvorträge<sup>27</sup> in Form von Steuerrückerstattungen<sup>28</sup> am Ende des Planungshorizontes aufgelöst.

Nach all diesen Modifikationen ergibt sich die effektive Steuerbelastung, die nun zur Beurteilung von Steuersystemen verwendet werden kann.

**Abbildung 10: "Vereinfachtes Ablaufschema der Modellrechnungen"**<sup>29</sup>

Eingabedaten

Bilanz- daten	Unter- nehmens- pläne	Gesamtwirt- schaftliche Daten	Anteils- eigner- daten	Steuer- daten	Wahl- rechte
------------------	-----------------------------	-------------------------------------	------------------------------	------------------	-----------------

Berechnungen



Erfolgs- mittlung	Vermögens- ermittlung	Steuerbe- rechnung	Liqui- dität	Bestandsfort- schreibung
----------------------	--------------------------	-----------------------	-----------------	-----------------------------

Ergebnisdaten



Steuer- zahlungen	Bilanz/GuV- Rechnungen	Liqui- dität	Endvermögen (Zielgröße)	Länderver- gleiche
----------------------	---------------------------	-----------------	----------------------------	-----------------------

Quelle: Spengel, C., *Steuerbelastungsvergleiche*, 1995, S. 165.

Ein grobes Ablaufschema ist in Abbildung 10 zu sehen, wobei die Ergebnisdaten der Vorperiode die Eingabeparameter der nächsten Periode darstellen.

<sup>26</sup> Bei einem linearen Tarif von 15 % und einem Vortrag um eine Periode in Höhe von einer Million € entspräche dies z. B. bei einem Zinssatz von 3 % ca. 145.631 € ( $1.000.000 \text{ €} \cdot 0.15 / 1.03$ ).

<sup>27</sup> Bei einem Bewertungsabschlag von z. B. 50 % wird nur noch die Hälfte der ursprünglichen Verlustvorträge steuermindernd berücksichtigt.

<sup>28</sup> Diese werden zur entsprechenden Abbildung im Modell genutzt. In der Praxis gibt es solche nicht.

<sup>29</sup> Spengel, C., *Steuerbelastungsvergleiche*, 1995, S. 163.

### 3 Anforderungen

Insbesondere die ständig wachsende Anzahl an Ländern erfordert zwingend die Wiederverwendung bereits modellierter Steuersysteme, da andernfalls die ständigen und notwendigen Aktualisierungen nicht mehr zu leisten wären. Dazu muss ein möglichst versionsunabhängiges und dauerhaftes Archiv für die Steuerinformationen einzelner Staaten geschaffen werden.

Des Weiteren soll eine effiziente Art der Dokumentation, die allen Mitarbeitern auf einfache Art und Weise zugänglich ist, bereit gestellt werden. Dadurch sollen Redundanzen, die aufgrund von mehrfacher Recherche und Validierung entstehen, vermieden werden. Daraus leiten sich direkt die nachfolgend erörterten Anforderungen an die Leistungen des Systems ab.

#### 3.1 Speicherung

Zur Vermeidung von Update-, Lösch- und Einfügeanomalien<sup>30</sup> ist es dringend erforderlich, jegliche redundante Datenspeicherung zu vermeiden. Wird ein Datum z. B. doppelt gespeichert, könnte bei einer Aktualisierung eine Kopie vergessen und damit der Datenbestand inkonsistent werden. Zudem muss sichergestellt werden, dass keine an sich unabhängigen Informationen direkt miteinander verknüpft sind, und getrennt erstellt und gelöscht werden können.

Bei der bisherigen Speicherung der Desktop-Version konnten konkurrierende Zugriffe quasi nur auf den Netzlaufwerken entstehen. Sollten zwei Anwender eine Datei gleichzeitig bearbeiten, tritt das Problem auf, dass die letzte Änderung persistiert wird und die vorhergehende verloren geht (Lost Update).<sup>31</sup>

Bei mehrbenutzerfähigen Datenbanken gibt es daher das Konzept von Transaktionen. Vereinfacht ausgedrückt, soll eine Transaktion alle Änderungen enthalten, die zwingend zusammengehören. Bei einer Überweisung zwischen zwei Konten sollten z.B. immer beide Kontostände aktualisiert werden und nicht nur einer von beiden. Des Weiteren müssen auch die zu Beginn der Transaktion eingelesenen Kontostände

---

<sup>30</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 176 f.

<sup>31</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 310.

noch gültig sein und nicht durch andere Vorgänge zwischenzeitlich verändert worden sein (Dirty Read).<sup>32</sup> Daraus leitet sich unmittelbar das ACID-Paradigma<sup>33</sup> ab:

- **Atomicity:** Die Transaktion soll nur die Operationen auf der Datenbasis enthalten, die zwingend zusammen ausgeführt werden müssen. Diese sollen entweder komplett oder gar nicht ausgeführt werden.
- **Consistency:** Die Datenbank muss sich nach Abschluss in einem konsistenten Zustand befinden. Es dürfen also keine Daten von nicht erfolgreichen Transaktionen enthalten sein.
- **Isolation:** Transaktionen dürfen sich nicht gegenseitig beeinflussen und müssen das exakt gleiche Resultat liefern, als wäre kein anderer Auftrag abgelaufen.
- **Durability:** Die Änderungen erfolgreicher Transaktionen müssen dauerhaft sein. Es muss garantiert sein, dass alle danach abgeschlossenen Teilaufträge diese Änderungen bereits eingelesen haben. Andernfalls müssen sie abgebrochen werden.

Zur Unterstützung der Mehrbenutzerfähigkeit ist es daher notwendig, dass die zu wählende Art der Speicherung das ACID-Paradigma unterstützt.

Bei einem konzeptionellen Entwurf kann es notwendig sein, die benötigten Daten aus den verschiedenen Perspektiven der Anwender zu modellieren. Diese müssen dann durch eine Sichtenintegration<sup>34</sup> zusammengeführt werden. Dabei können Namens- (Synonyme, Homonyme), Merkmals- (unterschiedliche Anzahl an Attributen), Struktur- (unterschiedliche Modellierung der gleichen Sachverhalte – z. B. Beziehung statt Objekt) und Abstraktionskonflikte<sup>35</sup> (in einer Sicht fehlen Daten) auftreten. Diese müssen vermieden werden.

Für diese Konflikte und obige Anomalien ist das Einhalten der ersten bis dritten Normalform (NF)<sup>36</sup> (formale Anforderungen an das Datenformat) zwingend erforderlich. Bis zu dieser NF kann die verlustfreie und abhängigkeitsbewahrende Zerle-

---

<sup>32</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 310 f.

<sup>33</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 283.

<sup>34</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 53 ff.

<sup>35</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 54 ff.

<sup>36</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 183 ff.

gung garantiert<sup>37</sup> werden, d. h. dass alle Attribute und funktionalen Abhängigkeiten (z. B. {Land, Steuerart, Jahr} → Tarif) erhalten bleiben.

### 3.2 Vereinfachte Zeitreihenanalyse

Zur Unterstützung der Zeitreihenberechnung muss garantiert werden, dass für jedes Land und jedes Jahr nur genau eine Version existiert. Diese Forderung würde aber jegliche Variationsrechnungen (z. B. Reformvorschläge) erschweren, da man diese Änderungen wieder rückgängig machen müsste. Daraus ergibt sich die Notwendigkeit der klaren Trennung von der aktuellen Rechtslage im jeweiligen Land zu einem bestimmten Zeitpunkt und sonstigen Variationen. Weiterhin gibt es auch häufig regionale Unterschiede bei der Besteuerung von Unternehmen (z. B. Gewerbesteuerhebesätze), sodass auch diese Möglichkeit berücksichtigt werden muss. Zudem ist die Vereinheitlichung der Ländernamen zumindest für die interne Darstellung erforderlich, da für die Datenbank andernfalls z. B. Holland und die Niederlande zwei verschiedene Länder wären.

Um die Qualität einer Modellierung zu dokumentieren, um dadurch die zeitintensive, erneute Überprüfung zu vermeiden, ist es notwendig, die Veränderungen zu dokumentieren und es dem Nutzer zu ermöglichen, die Qualität dieser Modifikation zu bestimmen. Damit auch in Zweifelsfällen eine effiziente Klärung möglich ist, ist es hilfreich zu dokumentieren, wer wann welche Änderung vorgenommen hat. Somit werden eine Benutzerverwaltung und eine Versionshistorie benötigt (vgl. 3.2.3.1 und 3.2.3.2).

#### 3.2.1 Dokumentation der Steuersysteme

Eine Überprüfung der Datengüte eines Steuersystems ist zurzeit auch deswegen so schwierig, weil die benutzten Quellen und die daraus abgeleiteten Annahmen häufig nicht mehr vorliegen und erneut recherchiert werden müssen. Dies stellt insbesondere aufgrund mangelnder Dokumentation und häufiger Fluktuation der Mitarbeiter ein Problem dar, da der Urheber der Modellierung oftmals nicht mehr verfügbar ist und eine exakte Dokumentation meistens nicht existiert.

---

<sup>37</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 195.

Deswegen soll das Archiv auch die Möglichkeit bieten, diese Quellen zu hinterlegen. Bisher wurde versucht, die Dokumentation mit einfachen Textkommentaren zu realisieren. Dies hat sich insbesondere in Bezug auf Onlinequellen als ungenügend erwiesen, da diese trotz Angabe einer Internetadresse sehr oft nicht mehr verfügbar waren.

Des Weiteren wird häufig nur die letztendliche Entscheidung dokumentiert und nicht der komplette Entscheidungs- oder Evaluationsprozess. So ist etwa die gewählte Abschreibungsart und -dauer einer Maschine ersichtlich. Nicht nachvollziehbar ist jedoch die Frage, warum andere Möglichkeiten nicht in Frage kommen und welcher Kategorie ein bestimmtes Wirtschaftsgut zugeordnet wurde. Sollten diese Regelungen geändert werden, sind diese Angaben notwendig, um eine auch im Zeitablauf einheitliche Einordnung vornehmen zu können. Somit muss dieser Prozess bei sich widersprechenden Quellen oder nicht erkenntlichen Annahmen in vielen Fällen erneut durchlaufen werden. Daher erscheint es notwendig, die Möglichkeit zu schaffen, auch komplexere Dokumente zu hinterlegen.

### 3.2.2 Recherche

Um den großen Datenmengen an Quellen gerecht zu werden und eine effiziente Entdeckung dieser Daten zu gewährleisten, ist eine praktikable Möglichkeit zur Recherche unabdingbar. Im Idealfall sollten alle benutzten Informationen hinterlegt werden. Dazu gehören z. B. Gesetzestexte, Richtlinien, Durchführungsverordnungen und Urteile. Auch selbst erstellte Dokumente, die den Entscheidungsprozess dokumentieren, sollten hinterlegt werden.

Für einzelne Details mag dies noch überschaubar sein. Durch die hohe Anzahl an Ländern und verschiedenen Jahrgängen wird dies aber sehr schnell nicht mehr handhabbar. Somit erscheint eine technische Unterstützung sinnvoll.

Zudem sollten die Quellen auch der konkreten Modellierung zugeordnet werden, so dass eine isolierte Speicherung auf Landesebene ohne jeglichen Bezug nicht sinnvoll erscheint. Dadurch sollte das Auffinden relevanter Dokumente auch deutlich erleichtert werden, wenn z. B. die Auswahl auf eine bestimmte Steuerart beschränkt werden kann.



### 3.2.3 Statusinformationen und Datenqualität

Um die Überprüfung der Qualität einer Modellierung auf ein Minimum zu reduzieren, ist es insbesondere bei der Bearbeitung durch mehrere Mitarbeiter wichtig, die Güte der Modifikationen zu dokumentieren. Hierdurch soll erreicht werden, dass es in Zukunft ausreicht, Bausteine, die noch nicht als validiert markiert worden sind, zu bearbeiten. Solange keine rechtlichen Änderungen einzupflegen sind, sollte somit jedes Land genau dann als überprüft gelten, wenn alle Bausteine validiert wurden.

#### 3.2.3.1 Benutzerverwaltung

Im Mehrbenutzerbetrieb besteht zudem der Bedarf, gewisse Rechte nur gewissen Mitgliedern einer Gruppe zur Verfügung zu stellen. So sollten z. B. Hilfskräfte nicht unbedingt das Recht haben, Modellierungen zu validieren. Zudem erscheint die Protokollierung des Nutzers einer Änderung zur raschen Klärung von Zweifelsfragen als sehr sinnvoll. Des Weiteren könnte für Recherchezwecke eine eigene Rolle vergeben werden, um durch den Entzug von Schreibrechten eine ungewollte Veränderung am Datenbestand zu verhindern. Deswegen erscheint es sinnvoll, eine Benutzerverwaltung zu etablieren.

#### 3.2.3.2 Versionshistorie

Weiterhin ermöglicht eine Versionshistorie, in Kombination mit einer Benutzerverwaltung, die Diskussion über durchgeführte Modifikationen, da nun Zeitpunkt und Bearbeiter der Änderung bekannt sind. Ein weiterer Vorteil ergibt sich aus der Möglichkeit, die Evaluationsgeschichte eines konkreten Landes nachzuvollziehen. Ist zum Beispiel der Rechtsstand 2008 aus der Kopie von 2007 und anschließenden Modifikationen hervorgegangen, ist es nun möglich, die genaue Ursache von Fehlern zu ergründen und diese bis zur Quelle zurück zu verfolgen. Solche Fehler müssen - falls vorhanden - in jeder abgeleiteten Version separat behoben werden.

Dies ist aufgrund der notwendigen, getrennten Datenspeicherung erforderlich, da sich andernfalls Änderungen an einem Rechtsstand unbeabsichtigt auch auf andere Rechtsstände auswirken würden. Daher können und müssen verschiedene Jahre einzelner Länder als selbstständige Länder angesehen werden. Die etwas schlechtere Möglichkeit der Fehlerbehebung muss unter diesen Prämissen in Kauf genommen werden.

### 3.2.4 Kontrollfunktion

Die bisherige Struktur der Steuermodellierung erwies sich aufgrund der gegebenen Komplexität, des Einsatzes von Formeln<sup>38</sup> und der sehr verschachtelten Darstellung von Dialogen als schwer validierbar. Neben einer Optimierung der Darstellung der Dialoge erscheint es daher angebracht, eine Funktion bereit zu stellen, die in einem recht übersichtlichen Format die wichtigsten Parameter einer Steuermodellierung eines Landes darstellt. Dadurch soll ein Abgleich mit den recherchierten Informationen erleichtert werden.

## 3.3 Nebenbedingungen

### 3.3.1 Verwendung ohne Anbindung an die Datenbank

Aufgrund der Tatsache, dass die Anbindung an ein zentrales Archiv nicht unter allen Umständen<sup>39</sup> zu gewährleisten ist, ist es notwendig, ein Arbeiten auch ohne direkte Verbindung zu ermöglichen. Hierfür wird eine Export- und Importfunktion zur temporären Zwischenspeicherung für die Desktopapplikation benötigt.

### 3.3.2 Performanz

Um die Nutzbarkeit einer solchen Lösung zu gewährleisten, ist ein gewisses Mindestmaß an Performanz notwendig. Dies betrifft nicht nur die reine Rechen- und Übertragungsgeschwindigkeit. Vielmehr ist auch die benötigte Dauer und Auslastung der Netzwerkverbindungen bei Client-Server-Architekturen<sup>40</sup> zu berücksichtigen. Ebenso ist der Bedarf an Hauptspeicher nicht zu vernachlässigen.

Daher muss neben der Beseitigung von möglichen Ineffizienzen auch ein Kompromiss zwischen den benötigten Ressourcen (Arbeitsspeicher, Netzwerklast, Übertragungs- und Rechengeschwindigkeit) gefunden werden, da die Optimierung der einen meistens zu Lasten der anderen geht.

---

<sup>38</sup> Vgl. Stetter, T., VITAX, 2005, S. 79.

<sup>39</sup> Z. B. bei Netzwerk-, Serverproblemen, wenn der Zugriff aus Sicherheitsgründen nur im Intranet möglich ist oder keine Verbindung zum Server möglich ist.

<sup>40</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 556.

### **3.4 Zusammenfassung**

Abschließend bleibt festzuhalten, dass ein Archiv für Steuerinformationen erstellt werden soll, welches die Zeitreihenanalyse und den Mehrbenutzerbetrieb unterstützt.

Des Weiteren sind die Implementierung einer Benutzerverwaltung und die automatische Pflege einer Versionshistorie gewünscht. Zur effizienten Validierung von Länderinformationen sollen ein Qualitätssystem und eine vereinfachte Kontrollmöglichkeit eingeführt werden.

Zur Vereinfachung der Recherche ist es erforderlich, alle benötigten Dokumente Kontext bezogen zu hinterlegen und deren einfache Suche zu ermöglichen.

## 4 Evaluation geeigneter Methoden zur Umsetzung

### 4.1 Auswahl der geeigneten Notation bzw. des Datenformates

Ein reines Archiv für Steuerinformationen mit elektronisch gespeicherten Quellen wäre nicht ausreichend für die hier gestellte Zielsetzung, da zum einen der zu erwartende Informationsgehalt wohl nur unwesentlich über bereits existierenden kommerziellen Lösungen<sup>41</sup> liegen würde. Zum anderen würde die Kenntnis, wie Daten für das Modell aufbereitet werden müssen,<sup>42</sup> bei diesem Ansatz verloren gehen. Zur Speicherung der Methodenkompetenz ist also ein Format zur Notation von Steuerinformationen notwendig. Zur weiteren Optimierung der Zeitreihenanalyse wäre es hilfreich, diese Notation möglichst direkt in das Analyseprogramm einlesen zu können.

#### 4.1.1 Bisheriges Datenmodell des European Tax Analyzer

Derzeit ist der European Tax Analyzer als bausteinbasiertes Metamodell<sup>43</sup> konzipiert. Dadurch wird der Benutzer in die Lage versetzt, Steuern in einfacher Art und Weise zu modellieren. Dies ermöglicht die Ausweitung des European Tax Analyzer auf eine Vielzahl von Ländern und Rechtsständen. Mittlerweile wurde der Wechsel von C/C++ zu Java vollzogen.

Bisher werden die Daten des Unternehmensplans, der steuerlichen Bewertungswahlrechte und –alternativen sowie der Modellierungen der einzelnen Steuerarten nicht getrennt gespeichert. Vielmehr wird sowohl für den Vorsteuer- als auch für den Nachsteuerfall jeweils eine separate Instanz des Datenmodells erzeugt.

Die Simulation der Steuerbelastungen ist nur bei identischen Ausgangsfällen<sup>44</sup> sinnvoll. Somit muss garantiert werden, dass diese Parameter für den Fall ohne und mit Steuern identisch sind. Für eine dauerhafte Speicherung ist die Lösung durch zwei isolierte Instanzen problematisch, da es zu einer redundanten Datenhaltung kommt.

---

<sup>41</sup> Vgl. IBFD, <http://www.ibfd.org/>, 10. 11. 2009.

<sup>42</sup> Vgl. Spengel, C., Taxation Knowledge Database, 2004, S. 8.

<sup>43</sup> Vgl. Stetter, T., VITAX, 2005.

<sup>44</sup> Ein Ausgangsfall bezeichnet hier ein Unternehmensmodell, das sich aus vielen Teilaspekten, wie z. B. dem Produktions- und Absatzplan und der Personal- und Gehaltsstruktur, zusammensetzt. Dies ist normalerweise für alle Länder identisch und garantiert somit eine wohl definierte Vergleichsbasis. Belastungsunterschiede sind somit nur durch rein steuerliche Aspekte induziert.

Beim Laden, Ändern und vor jeder Berechnung erfolgt eine manuelle<sup>45</sup> Synchronisation der beiden Instanzen. Dadurch wird die doppelte Speicherung dieser Daten in Kauf genommen.

Bisher ist der Wechsel des Unternehmensmodells nur durch das Kopieren aller notwendigen Parameter möglich. Für die längerfristige Speicherung wird daher die Trennung der Unternehmensdaten von den Daten für die Steuermodellierung vorgeschlagen.

Um Lösch- und Updateanomalien<sup>46</sup> auszuschließen, sollte die bisherige Organisation der Daten geändert werden. Des Weiteren erscheint eine Anpassung der temporären Datenhaltung an diese Modifikationen sinnvoll.

Daher wird die Spaltung in Vorsteuer- und Nachsteuerunternehmen aufgehoben und die Datenmodelle vereinigt. Die zu verwaltenden Daten lassen sich grob in vier Kategorien aufteilen:

- Steuerlich unabhängige Daten (z. B. Produktions- und Absatzplan);
- Steuerlich abhängige Daten (z. B. Verbrauchsfolgeverfahren und ähnliche Wahlrechte);
- Steuerdaten, die nur zur Ermittlung der Steuern benötigt werden;
- Ergebnisdaten.

Die Ergebnisdaten lassen sich aus den anderen Daten berechnen und bedürfen daher keiner Speicherung.

Bisher ist es möglich, einzelnen Ländern ein „eigenes“ Vorsteuermodell zuzuweisen.<sup>47</sup> Dies soll auch grundsätzlich erhalten bleiben. Im Standardfall, bei dem jedes Land das gleiche Vorsteuermodell benutzt, wird in der neuen Struktur über eine Objektreferenz auf die Daten des sogenannten „Basismodells“ verwiesen. Nur bei einem

---

<sup>45</sup> Manuell meint hier, dass es notwendig ist, explizit Programmcode dafür bereitzustellen und zu warten.

<sup>46</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 150 f.

<sup>47</sup> „Die von Eckerle mit dem European Tax Analyzer verfolgte Zielkonzeption der Analyse der steuerlichen Einflussnahme auf die Investitionsentscheidung von Unternehmen mit ländertypischen Unternehmensstrukturen und deren Umsetzung sowie Ausgestaltung eignet sich als Ausgangspunkt für die dieser Untersuchung zugrunde liegende Zielsetzung nicht. Denn Steuerwirkungsanalysen sind nur bei Verwendung identischer Ausgangsdaten vor Steuern für die entsprechenden Modellunternehmen aussagekräftig.“ Gutekunst, G., Steuerbelastungen, 2005, S. 65 Fußnote 209. Vgl. außerdem Eckerle, T. H., Standortunterschiede, 2000.

separaten Vorsteuermodell werden eigene Instanzen der Kategorie I und der Vorsteuervariante der Kategorie II erstellt. Dadurch wird eine Datenduplizierung vermieden.

Für die Wahl des Datenformates bzw. der Notation hat dies keinen Einfluss, da nur durch steuerliche Regelungen beeinflusste Daten gespeichert werden sollen. Nur Kategorie II muss sowohl Bestandteil der ökonomischen als auch der steuerlichen Daten sein. Die strikte Trennung zwischen steuerlichen und nicht steuerlichen Daten führt jedoch zu dem Problem, dass Referenzen auf Teile des Unternehmensmodells rekonstruiert werden müssen. So greift z. B. die Grundsteuer i. d. R. auf Daten der einzelnen Grundstücke zurück.

Durch die getrennte Speicherung gehen die zur Laufzeit des Programms eindeutigen Objektreferenzen verloren und müssen daher ersetzt werden. Dies betrifft nicht nur Bilanzobjekte und Anteilseignerdaten, sondern auch Zwischenergebnisse, wie z. B. die einzelnen Posten aus der Gewinn-und-Verlust-Rechnung.

Als Lösung werden eindeutige Namen als Referenzen benutzt. Hierfür bieten sich in Java Enums<sup>48</sup> (Aufzählungen) an. Diese werden zur Laufzeit aufgelöst und durch die entsprechenden Objektreferenzen ersetzt. Dies erhöht zwar die Komplexität, jedoch ist nur so die getrennte Speicherung möglich. Diese Trennung ist unabdingbar, da die beiden Komponenten austauschbar sein müssen, um eine Vielzahl von Variationen zu ermöglichen. Denn es muss möglich sein, die gleiche Steuermodellierung auf eine große Anzahl verschiedener Unternehmensmodelle anzuwenden und umgekehrt.

#### 4.1.2 Aktuelles Konzept der Datenspeicherung

In der bisherigen Version des European Tax Analyzer wird das Datenmodell im Extensible Markup Language (XML)<sup>49</sup> Format gespeichert. Die verwendete XML-Speicherung aus dem *java.beans*-package basiert auf dem Konzept der Persistenz durch Erreichbarkeit.<sup>50</sup> Dies hatte den Vorteil, dass jedes Objekt, das über einen Pfad an Referenzen erreichbar ist, gespeichert wird, wenn es gewissen Anforderungen ge-

<sup>48</sup> Seit Java 5.0 (1.5) im Sprachumfang. Vgl. Sun Microsystems Inc., Enums, <http://java.sun.com/.../enums.html>, 29. 01. 2009.

<sup>49</sup> Dies wurde mithilfe der Klassen *java.beans.XMLEncoder* und *java.beans.XMLDecoder* aus der Java-Standard-Bibliothek (J2SE) erledigt.

<sup>50</sup> Vgl. Milne, P., XMLEncoder, <http://java.sun.com/.../persistence4/>, 09. 02. 2009.

nügt. Primär bedeutet dies die Bean-Eigenschaften<sup>51</sup> zu erfüllen. Diese Anforderungen werden häufig bereits implizit erfüllt, sodass ein sorgfältiges Design notwendig ist, um nur die benötigten, nicht redundanten Daten zu speichern.

Inkonsistenzen des Datenformats zu entdecken ist besonders schwierig, da bei der ersten Konzeption i. d. R. keine Probleme auftreten. Im Normalfall wird eine Art Abbild der aktuell gültigen Parameter erstellt und wieder eingelesen. Die Speicherung von redundanten Daten oder Ergebnissen fällt hierbei nicht auf, solange deren Behandlung im Programm nicht verändert wird. Erst bei Modifikationen werden diese Probleme unter Umständen sichtbar, sodass solche Inkonsistenzen häufig erst sehr spät entdeckt werden.

Um den benötigten Speicherbedarf zu minimieren, speichern die Routinen der Standardimplementierung nur die im Vergleich zu den Standardwerten<sup>52</sup> geänderten Attribute. Solange die aktuelle Version nicht geändert wird, ist dies unproblematisch und auf den ersten Blick vorteilhaft.

Bei einer Änderung eines solchen Standardwertes gilt dieser nun aber auch für bereits gespeicherte Modelle und kann zu unerwünschten Nebeneffekten führen. Bei einer Änderung des Standardsatzes, z. B. für die Pool-Abschreibung<sup>53</sup> von 20 % auf 25 %, sind die alten Resultate nicht mehr nachvollziehbar: Dies begründet sich damit, dass der erste Wert von 20 % (alter Normwert) nicht gespeichert worden ist und daher nun automatisch mit 25 % gerechnet wird.

Insbesondere ist Vorsicht bei der Speicherung von Sammlungen<sup>54</sup> von Objekten (z. B. in Form von Listen) geboten, da hier evtl. neue Standardelemente beim Laden wieder entfernt werden. Da die *set*-Methode<sup>55</sup> nicht explizit zur Rekonstruktion benutzt wird, ist somit auch ein expliziter Eingriff nicht möglich. Daher bestehen ohne

---

<sup>51</sup> Dies erfordert primär die Implementierung eines argumentlosen Konstruktors und die Bereitstellung von Zugriffs- und Änderungsmethoden für alle Attribute. Zudem müssen Namenskonventionen beachtet werden. Die Zugriffsfunktion muss z. B. immer aus „*get*“ plus Attributnamen bestehen.

<sup>52</sup> Werte, die die Instanz nach dem Erzeugen durch den argumentlosen Konstruktor besitzt.

<sup>53</sup> Die Pool-Abschreibung setzt jede Periode einen festgelegten Prozentsatz vom Restbuchwert als Aufwand für die Abnutzung eines Wirtschaftsgutes an. Vgl. auch Kapitel 2.2.1.

<sup>54</sup> Hiermit sind alle Klassen gemeint, die das Interface *java.util.Collection* (z. B. *Set*, *List* oder *Map*) implementieren.

<sup>55</sup> I. d. R. wird die Kombination aus *get*, *clear* und *put* oder *add* benutzt.

spezielle Vorkehrungen kaum Möglichkeiten, solche unerwünschten Nebeneffekte zu vermeiden.

Der Vorteil der Fehlertoleranz bzgl. beschädigter Dateien<sup>56</sup> hat die Kehrseite, dass ein teilweiser Defekt i. d. R. auch zu einem inkonsistenten Datenmodell führt. Somit sind auch die noch rekonstruierbaren Daten mit hoher Wahrscheinlichkeit unbrauchbar, bzw. müssten explizit um fehlende Informationen ergänzt werden.

Zunächst erscheint es sinnvoll, dass einfache Änderungen an einer Klasse (z. B. das Entfernen einer Eigenschaft) die ältere gespeicherte Version nicht unbedingt unbrauchbar machen. Dies wird dadurch erreicht, dass bei der nächsten Speicherung dieses Attribut nicht mehr serialisiert wird und somit ein Art Reparatur<sup>57</sup> stattfindet. Es handelt sich jedoch um eine potentielle Fehlerquelle, die explizit behandelt werden sollte. Das Umbenennen eines Attributs würde sogar dazu führen, dass der alte Wert ignoriert und bei der nächsten Speicherung unbemerkt<sup>58</sup> auf den Standardwert zurückgesetzt würde.

Die Restriktion bzgl. Veränderungen ist nicht nur auf Modifikationen des Datenschemas beschränkt.<sup>59</sup> Sie beeinflusst aufgrund von diversen Restriktionen<sup>60</sup> auch Änderungen an der Geschäftslogik. Somit hat nahezu jede Änderung am Programmcode (z. B. Wechsel eines Standardwertes) auch Einfluss auf das gespeicherte Datenformat. Daher ist dieses Konzept für die dauerhafte Speicherung weniger geeignet.

Für die nicht auf Steuern bezogenen Daten soll die XML-Serialisierung weiterhin für die permanente persistente Datenhaltung verwendet werden, wobei hierbei besonders auf obige Probleme geachtet werden muss. Bei der Verwendung des Immutab-

---

<sup>56</sup> Vgl. Stetter, T., VITAX, 2005, S. 170.

<sup>57</sup> Beim Entfernen eines Attributs würde beim Einlesen einer älteren Version eine Ausnahme erzeugt. Die Informationen zu diesem Attribut gehen durch die erneute Speicherung verloren, sodass dieser Fehler nicht mehr auftritt.

<sup>58</sup> Beim Einlesen wird zwar eine Ausnahme erzeugt, die aber nicht sinnvoll behandelt werden kann, so dass dies dem Anwender normalerweise verborgen bleibt.

<sup>59</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 19.

<sup>60</sup> Insbesondere die geforderten Bean-Eigenschaften und das Rekonstruktionsverhalten bei Collection-Elementen seien hier erwähnt. Auf die Verwendung von `java.beans.PersistenceDelegate` wurden bisher nicht verwendet. Vgl. Milne, P., XMLEncoder, <http://java.sun.com/.../persistence4/>, 09. 02. 2009.



le-Patterns (unveränderbare Klasse)<sup>61</sup> muss explizit auf eine Speicherung durch Persistence Delegates (explizite Anweisungen zur Speicherung) geachtet werden.

Es empfiehlt sich daher, durch gutes Design des gespeicherten Datenmodells dafür zu sorgen, dass der Bedarf für solche Änderungen möglichst gering und unwahrscheinlich ist.

Um den Wunsch der Anwender, die Daten ohne Zugang zum Archiv nutzen zu können zu erfüllen, wird diese auch zum temporären Export von Steuerdaten genutzt. Die persistente Datenspeicherung soll jedoch ausschließlich über das Archiv erfolgen. Somit entfällt die Notwendigkeit einer versionsübergreifenden Kompatibilität dieser Funktion. Die temporäre lokale Zwischenspeicherung erhöht somit primär den Komfort für den Anwender und muss daher geringere Anforderungen erfüllen als zuvor. Deswegen ist es nicht erforderlich, besondere Vorkehrungen zu treffen.

Ein besonderer Vorteil dieser temporären Export-Möglichkeit besteht darin, dass nun zwei Arten der Speicherung existieren. Dadurch werden auch Reorganisationen des für die längerfristige Speicherung vorgesehenen Formats möglich. Solange die temporäre Speicherung unverändert bleibt, kann eine Konversion über das Programm erfolgen. Dies geschieht durch den Import der vorher exportierten Daten in das modifizierte Format. Dieser Mechanismus kann auch zur Portierung zwischen verschiedenen Softwareversionen oder Softwareanbietern<sup>62</sup> benutzt werden. Somit werden die Kosten für eine solche Reorganisation drastisch reduziert und dabei die Flexibilität erhöht.

## 4.2 Mehrbenutzerbetrieb

Aufgrund der Anforderung, die Benutzung des Archivs auch bei mehreren gleichzeitigen Zugriffen von verschiedenen Nutzern zu ermöglichen, ist eine Unterstützung des ACID-Paradigmas (vgl. 3.1) zwingend erforderlich. Datenbankmanagementsysteme (DBMS) erfüllen diese Voraussetzung bereits, sodass es sinnvoll erscheint, ein solches zu nutzen.

---

<sup>61</sup> Die Werte der Eigenschaften einer Instanz einer Klasse können nach deren Erzeugung nicht mehr geändert werden. Wird eine Instanz mit anderem Inhalt benötigt, muss eine neue Instanz erzeugt und gegen die alte ausgetauscht werden. Vgl. Bloch, J., Effective Java, 2008, S. 73 ff. Item 15.

<sup>62</sup> Sollte zum Beispiel eine Datenbank zur Anwendung kommen, könnte das Datenbankmanagementsystem auf einfache Art und Weise bei Bedarf ausgetauscht werden.

Würde die Datenbankanbindung nur exklusiven Zugriff bieten, wäre eine gleichzeitige Benutzung nahezu ausgeschlossen. Somit könnte ein Engpass entstehen. Daher muss das DBMS und die darauf zugreifenden Programme für einen Mehrbenutzerbetrieb ausgelegt sein bzw. damit umgehen können. Zur Realisierung eines solchen Mehrbenutzerbetriebs bieten sich grundsätzlich zwei Strategien an.

Neben der Verwendung von Transaktionen, die es erlauben temporäre Änderungen am Ende explizit zu verwerfen oder zu speichern, muss verhindert werden, dass zwei Transaktionen dasselbe Objekt ändern. Hierfür werden i. d. R. Sperren benutzt. Hierbei wird zwischen einer pessimistischen und einer optimistischen Sperrstrategie unterschieden. Beide Strategien haben aber ihre Vor- und Nachteile. Welche Vor- und Nachteile stärker zum Tragen kommen und welcher Strategie damit der Vorzug gegeben werden sollte, hängt sehr stark von dem zu erwartenden Lastprofil der Datenbank ab und der Frage, wie hoch eventuelle Datenverluste<sup>63</sup> zu bewerten sind.

Die pessimistische Strategie hat den Vorteil, dass quasi keine Datenverluste auftreten können, da die Bearbeitung sowie i. d. R. sogar das Lesen des gesperrten Objektes von vornherein verhindert wird. Die optimistische Strategie erlaubt dies prinzipiell zunächst einmal. Nur in dem Fall, dass die Quelle zwischenzeitlich geändert wurde, wird die Änderung abgelehnt. Dies hat den Vorteil, dass bei primär lesendem Zugriff eine höhere Verfügbarkeit gewährleistet ist, da andernfalls gesperrte Objekte nun lesbar sind und eine Überprüfung von Sperren entfällt. Das Problem des Datenverlustes kann zum einen durch die Erhöhung der Granularität der Sperren reduziert werden, zum anderen ist es möglich, die Änderungen in der lokalen Anwendung temporär zwischenspeichern und dem Anwender die Konfliktlösung explizit zu übertragen. Er muss dann entscheiden, welche Änderungen die Richtigen sind. Zudem hat er die Möglichkeit mit dem Kollegen Rücksprache zu halten, der die vorherigen Änderungen vorgenommen hat.

Das optimistische Verfahren wird präferiert, da es eine bessere Performanz und Skalierbarkeit ermöglicht. Zudem wird die Wahrscheinlichkeit einer gleichzeitigen Modifikation eines Landes durch mehrere Mitarbeiter, aufgrund der zurzeit vorherr-

---

<sup>63</sup> Datenverlust bedeutet hier, dass aktuelle Änderungen evtl. nicht gespeichert werden können, da zwischenzeitlich von einem anderen Benutzer andere Änderungen vorgenommen wurden.

schen Arbeitsteilung, als sehr gering eingeschätzt. Somit sollten Datenverluste äußerst selten auftreten. Aus diesem Gründen erscheint diese Methode als besser geeignet.

### 4.3 Anforderungen an das Datenbankschema

„Ein Schema ist eine in einer Datenbeschreibungssprache abgefaßte [sic!] Definition der in einer Datenbank zugelassenen Datenstrukturen.“<sup>64</sup> Es „legt die Struktur der abspeicherbaren Datenobjekte fest. [...] Deshalb kann man das Datenbankschema auch als Metadaten – also Daten über Daten – verstehen.“<sup>65</sup>

Aufgrund der Tatsache, dass jegliche längerfristige Speicherung dazu führt, dass gewisse Teile des Quellcodes nicht oder nur noch mit sehr hohem Aufwand veränderbar sind,<sup>66</sup> ist es prinzipiell notwendig, diese sorgfältig zu designen. Die Umsetzung einer persistenten Speicherung in Java<sup>67</sup> benötigt auf den ersten Blick nur wenig Aufwand. Daher wird häufig übersehen, dass eine langfristige Speicherung deutlich mehr Aufwand erfordert.<sup>68</sup>

Hierzu ist es in erster Linie erforderlich, die vorhandenen Daten entsprechend der Kategorisierung von oben (vgl. 4.1.1) zu unterteilen und danach entsprechend zu separieren. Die erste Kategorie besteht dabei aus den Eingabeparametern des Unternehmensmodells und allen sonstigen Parametern, die unabhängig von der Steuerermittlung sind. An zweiter Stelle stehen die steuerspezifischen Parameter. Die letzte Kategorie besteht aus temporären und ableitbaren Daten und darf nicht gespeichert werden, da sie zur Verfälschung von Rechenergebnissen führen könnte. Die zweite Kategorie ist nochmals zu unterteilen.<sup>69</sup> Des Weiteren ist zu überprüfen, inwieweit die Speicherung zu trennen ist. Dies ist vor allem der Tatsache geschuldet, dass die Steuermodellierung nicht abhängig von einer genauen Ausprägung eines Unternehmensmodells sein sollte.

---

<sup>64</sup> Engesser, H./Claus, V., 1993, S. 161.

<sup>65</sup> Kemper, A./Eickler, A., 2009, S. 24.

<sup>66</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 31.

<sup>67</sup> Trademark von Sun Microsystems, Inc.

<sup>68</sup> Vgl. Bloch, J., Effective Java, 2008, S. 289.

<sup>69</sup> Siehe Kapitel 4.1.1

Aufgrund der bereits existierenden Anwendung erscheint es nicht sinnvoll, ein komplett neues Datenbankschema mit zugehörigen Berechnungs- und Ausgabemodulen zu entwerfen, sodass dies im Rahmen einer Refaktorisierung des Quellcodes durchgeführt werden soll.

Die Klassenhierarchie soll durch Generalisierung und Spezialisierung bzw. Aggregation<sup>70</sup> optimiert werden, um ein möglichst gutes Datenbankschema zu garantieren. Somit können auch Namens-, Merkmals-, Struktur- und Abstraktionskonflikte (vgl. 3.1) vermieden werden.

#### **4.4 Performanz**

Neben der Wahl der Sperrstrategie und der zugrunde liegenden Infrastruktur ist es für die Performanz einer Datenbank wichtig, das Datenschema zu optimieren, um die Anzahl benötigter Verknüpfungen von Tabellen durch Kreuzprodukte (Joins)<sup>71</sup> für eine Datenbankabfrage möglichst zu minimieren. Denn zur vollständigen Rekonstruktion eines Objektes müssen häufig die Daten aus verschiedenen Tabellen zusammengetragen werden. Ohne entsprechende Optimierung<sup>72</sup> einer solchen Anfrage müssen alle gleichartigen Objekte betrachtet werden. Hierfür wird i. d. R. eine temporäre Tabelle mit den Kreuzprodukten der Untertabellen angelegt. Durch mehrwertige Beziehungen (1:n, n:1 oder n:m) erhöht sich die Anzahl an Zeilen, da für jede Kombination aus dem Primärschlüssel (der ersten Tabelle) und dem identischen Fremdschlüssel (Primärschlüssel der ersten Tabelle in der zweiten Tabelle) eine separate Zeile hinzugefügt werden muss. Aus diesem Grunde müssen die bereits vorhandenen Klassen optimiert und die Speicherung redundanter Daten verhindert werden. Insbesondere ist bei 1:n Beziehungen möglichst auf eine Hinterlegung auf der

---

<sup>70</sup> Kemper, A./Eickler, A., 2009, S. 49 ff.

<sup>71</sup> Auch als kartesisches Produkt bezeichnet.

<sup>72</sup> Neben der Verwendung von speziellen Arten von Joins (Vgl. Kemper, A./Eickler, A., 2009, S. 94.) kann auch die Aufspaltung einer Abfrage in mehrere Teilabfragen sinnvoll sein. Bei einer solchen Aufteilung liegt der Vorteil darin, dass mithilfe des vorherigen Ergebnisses die Zahl der infrage kommenden Datensätze stark reduziert werden kann.

N-Seite<sup>73</sup> zu achten, da ansonsten eine zusätzliche Verknüpfungs-Tabelle (Join-Table)<sup>74</sup> benötigt wird.

Des Weiteren sind auch die einzelnen Abfragen daraufhin zu optimieren, dass möglichst nur benötigte Informationen angefordert werden. Dies bezüglich gilt es abzuwägen. Eventuell kann es sinnvoll sein, trotzdem größere Datenmengen auf einmal zu laden, wenn der Nutzer diese mit hoher Wahrscheinlichkeit im nächsten Schritt sowieso benötigen wird. Hierbei sollte bei der Spezifizierung von Beziehungen zwischen einzelnen Objekten darauf geachtet werden, dass diese Verweise nicht zwangsweise zum (Nach-) Laden<sup>75</sup> der referenzierten Objekte führen, sofern diese nicht direkt voneinander abhängen. Andernfalls könnte dies leicht dazu führen, dass immer eine sehr große Menge an Daten geladen wird, obwohl sie nicht, nur zum Teil oder noch nicht benötigt wird. Somit würden die benötigten Datenmengen unnötig ansteigen und die Reaktionszeit auf Benutzeranfragen könnte inakzeptabel lang werden.

#### 4.5 Dokumentation der Steuersysteme

Zur effizienten Dokumentation erscheint es sinnvoll, möglichst alle denkbaren Quellen nutzen zu können. Aus Vereinfachungsgründen werden diese auf das PDF-Format<sup>76</sup> beschränkt. Dies stellt jedoch keine Einschränkung an sich dar, da das Format ein hohes Maß an Flexibilität bietet und auch von den verfügbaren Scannern erzeugt werden kann. Darüber hinaus hat sich dieses Format als Standard<sup>77</sup> für die Langzeitarchivierung und den Dokumentenaustausch im Internet etabliert. Somit sollte nahezu jede Quelle hinterlegbar und auch in Zukunft noch lesbar sein. Dokumente, die sich weder drucken (und anschließend scannen) noch in diesem Format speichern oder zu einer solchen Datei konvertieren lassen, können mithilfe von Screenshots umgewandelt werden, auch wenn dies bei längeren Dokumenten sehr

---

<sup>73</sup> Aufgrund der Tatsache, dass bei solchen Beziehungen jedes Objekt nur genau einmal referenziert werden kann, reicht es aus den Fremdschlüssel der 1-Seite in die Tabelle als Spalte aufzunehmen.

<sup>74</sup> In einer solchen Tabelle stehen i. d. R. nur die beiden Fremdschlüssel der verknüpften Tabellen. Bei Beziehungen mit definierter Reihenfolge (z. B. aufsteigend sortiert) wird zusätzlich eine Indexspalte benötigt.

<sup>75</sup> In Hibernate-Annotations sind dafür die Attribute „*lazy*“ und „*eager*“ vorgesehen.

<sup>76</sup> Portable Document Format (PDF) und ist eine Trademark von Adobe Systems Incorporated.

<sup>77</sup> „Die International Organization of Standardization (ISO) hat PDF/Archive als Standard für die Langzeitarchivierung von Dokumenten anerkannt: ISO 19005-1 stützt sich auf das von Adobe entwickelte und weitgehend offen dokumentierte Portable Document Format (PDF).“ Trinkwalder, A., PDF/A als ISO-Standard, <http://www.heise.de/.../63957>.

mühsam sein kann. Letzterer Fall ist jedoch eher hypothetischer Natur und daher nicht entscheidungsrelevant.

Um einen möglichst großen Kontextbezug zu ermöglichen, erscheint es nicht sinnvoll, die Dokumentation auf die Steuerart zu beschränken. Vielmehr sollten auch einzelne Teile einer Steuer zur Aufnahme von Dokumenten fähig sein. Aufgrund der Tatsache, dass jeder Unterbaustein einer Steuer auch einen Tarif enthält,<sup>78</sup> ist es sinnvoll, die Verknüpfung dieses Unterbausteins mit den benötigten Quellen über den Tarif zu ermöglichen. Die Dokumentation der Steuerart sollte beim tatsächlichen Tarif erfolgen. Dies stellt somit einen guten Kompromiss zwischen größtmöglicher Granularität der Dokumentation und einer möglichst einfachen Umsetzung dar. Um zu vermeiden, dass das Löschen eines solchen Tarifes oder des gesamten Sammelbausteins auch den Verlust aller Quellen mit sich zieht, werden die Quellen auch automatisch mit dem jeweiligen Ländermodell verknüpft. Sollte nachträglich eine explizite Löschung gewünscht sein, kann dies über die Liste im Ländermodell erreicht werden.

Für Wahlrechte (z. B. Abschreibungen oder die Verbrauchsfolge), die die Ermittlung des Gewinns beeinflussen und somit nicht direkt einer Steuerart zuzurechnen sind, ist die Dokumentation auf Länderebene vorgesehen. Diese werden im Programm gesondert hinterlegt, sodass auch bei Ländern mit nur einer gewinnabhängigen Steuerart eine Hinterlegung im Tarif nicht sinnvoll ist.<sup>79</sup>

Eine noch höhere Granularität (z. B. auf Ebene der Bemessungsgrundlagenbausteine) erscheint aus Vereinfachungsgründen nicht sinnvoll. Es besteht jedoch auch weiterhin die Möglichkeit, auf jeder Ebene Textkommentare zu hinterlegen.

## 4.6 Objektorientierte vs. relationale Datenbank

Eigentlich liegt es sehr nahe, für ein Programm, das in einer objektorientierten Sprache geschrieben wurde, auch eine solche Datenbank auszuwählen. Der primäre Vor-

---

<sup>78</sup> Sollte ein Tarif nicht notwendig sein, wird ein linearer Tarif mit 100 % verwendet. Vgl. auch Kapitel 2.2.1.

<sup>79</sup> Bei Ländern mit nur einer gewinnabhängigen Steuerart könnten die Regeln zur Gewinnermittlung im Grunde genommen dieser zugeordnet werden. Im Programm werden solche Informationen aber losgelöst von der genauen Steuermodellierung spezifiziert, sodass eine Hinterlegung der Regelungen z. B. zu den Abschreibungen im Steuertarif trotzdem nicht sinnvoll ist.

teil liegt darin, dass man die Daten 1:1 übernehmen kann und diese nicht erst durch eine Abbildung (Mapping) auf relationale Tabellen (Relation)<sup>80</sup> transformiert werden müssen. Durch eine solche Abbildung können folgende Probleme<sup>81</sup> (auch als „Impedance Mismatch“<sup>82</sup> bezeichnet) auftreten:

- **Datentypen:** Häufig bestehen unterschiedliche Konventionen bezüglich der Größe und des Formats eines Datentyps. Eine Zeichenkette wird z. B. in Java unabhängig von ihrer Länge i. d. R. als *String* definiert. In der Datenbank werden kurze Folgen aber als *VARCHAR* und längere als *CLOB* (Character Large Object) gespeichert.
- **Segmentierung:** Bei der Abbildung von Klassen auf Relationen ist es häufig notwendig, diese auf mehrere Tabellen aufzuteilen. „Bei einem Zugriff auf ein segmentiertes Objekt muß [sic!] dieses mittels Verbund (also Join-) Operationen (mühsam und zeitaufwendig) zusammengebaut werden.“<sup>83</sup>
- **Identität und künstliche Schlüsselattribute:** In Java müssen Objekte nur zur Laufzeit des Programms eindeutig sein und besitzen eine Art künstliche Identität. Daher ist es auch möglich, zwei identische Instanzen mit exakt den gleichen Attributwerten zu erzeugen. Für eine Datenbank dürfen keine zwei deckungsgleichen Instanzen existieren. Zudem müssen sie auch über das Programmende hinaus eindeutig bleiben. Deswegen muss ein Schlüssel (ID) einzigartig sein. Gibt es keinen natürlichen Schlüssel (z. B. der Ländernamen),<sup>84</sup> muss ein künstliches Schlüsselattribut eingefügt werden.
- **Fehlendes Verhalten:** Anwendungsspezifisches Verhalten (z. B. eine Funktion, um aus dem Geburtsdatum das Alter zu berechnen) wird in relationalen Schemas nicht berücksichtigt.

---

<sup>80</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 20 ff.

<sup>81</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 21; Kemper, A./Eickler, A., 2009, S. 373.

<sup>82</sup> Kemper, A./Eickler, A., 2009, S. 374.

<sup>83</sup> Kemper, A./Eickler, A., 2009, S. 373.

<sup>84</sup> Im konkreten Fall bildet der Ländername nur einen Teil eines zusammengesetzten natürlichen Schlüssels. Dieser besteht für das Ländermodell aus dem Namen, dem Jahr, der Variation und einem erläuternden Kommentar.

- **Kardinalitäten und Attributmengen:** Mehrwertige Beziehungen<sup>85</sup> zwischen Objekten werden in der Regel durch Listen abgebildet. Bei 1:n-Beziehungen kann sich z. B. die Richtung der Beziehung in der Datenbank umdrehen. Bei einem Verzicht auf eine separate Verknüpfungstabelle (Join-Table) wird dann in der Tabelle der von der Liste verwalteten Objekte der Schlüssel (Fremdschlüssel) des Objektes mit der Liste eingetragen. In Java ist die Beziehung also auf der Seite des Listenverwalters zu sehen, während sie in der Datenbank nur in der Tabelle der verwalteten Objekte erkennbar ist. Besonders n:m-Beziehungen<sup>86</sup> müssen gesondert behandelt werden, da ansonsten jede Seite der Beziehung getrennt verwaltet werden würde.
- **Vererbung:** Dieses Konzept ist relationalen Datenbanken unbekannt, da sie nur das Fremdschlüsselkonzept, das einer Komposition (Beziehung) entspricht, kennen. Es gibt jedoch Strategien, die eine indirekte Abbildung erlauben und den polymorphen Zugriff (Objekte mit gleicher Schnittstelle können einheitlich behandelt werden) gewährleisten.

Diese Strategien<sup>87</sup> zur Abbildung von Vererbungshierarchien sind:

- Eine Tabelle pro Klassenhierarchie („Single Table“);
- Eine Tabelle je konkreter Klasse („Table per Class“);
- Oder eine Tabelle je Klasse („Joined Table“).

Bei der ersten Alternative („**Single Table**“) werden alle Klassen einer Hierarchie in einer einzigen Tabelle abgebildet. Durch eine separate Spalte wird festgehalten, zu welcher Klasse die Daten gehören. Die Tabelle muss somit alle möglichen Attribute enthalten. Bei sehr unterschiedlichen Attributmengen kann dies dazu führen, dass sehr viel Speicherplatz für nicht benutzte Eigenschaften verschwendet wird. Es muss

---

<sup>85</sup> Eine Person hat z. B. ein Geburtsdatum. Dies entspräche einer 1:1-Beziehung. Sie kann mehrere Vornamen (1:n) besitzen. Alle Personen können beliebig viele andere Personen (n:m) kennen. 1:n und n:m werden als mehrwertige Beziehungen bezeichnet.

<sup>86</sup> Mit Rücksicht auf die Funktionsweise des Garbage-Collectors (Speicherreinigungsroutine) sollten n:m-Beziehungen in Java wenn möglich vermieden werden. Durch die doppelten Rückreferenzen kann die Routine kaum noch feststellen, ob das Objekt noch gebraucht wird oder nicht. Somit ist eine explizite Behandlung zu empfehlen, falls eine solche Beziehung unbedingt gebraucht wird. Daher treten primär nur 1:1-Beziehungen in Form von Kompositionen (Attribut enthält eine Referenz auf eine konkrete Instanz) und 1:n-Beziehungen auf, die i. d. R. durch Sammlungen wie z. B. Listen abgebildet werden.

<sup>87</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 82 ff.



jedoch immer nur eine Tabelle gelesen werden, sodass eine gute Performanz zu erwarten ist.

„**Table per Class**“ erzeugt für jede konkrete Klasse (nicht für abstrakte) eine eigene Tabelle. Dadurch werden teilweise nicht genutzte Spalten vermieden. Bei einer polymorphen Abfrage müssen jedoch nun alle Tabellen eingelesen werden. Des Weiteren besteht bei gemeinsamen Attributen eine gewisse Redundanz.<sup>88</sup> Zudem muss garantiert werden, dass der Primärschlüssel über alle Tabellen hinweg eindeutig ist.

„**Joined Table**“ hingegen lagert nur zusätzliche Attribute in zusätzliche Tabellen aus. Hierbei wird jegliche Redundanz und Platzverschwendung<sup>89</sup> verhindert und ein normalisiertes Datenbankschema gewährleistet. Die Performanz leidet jedoch unter den vielen erforderlichen Verknüpfungen der für die Rekonstruktion der Objekte notwendigen Tabellen.

Welche Strategie am vorteilhaftesten ist, muss für jeden Einzelfall gesondert geprüft werden. Nicht jede Variante garantiert die Einhaltung der formalen Vorschriften, die von Normalformen gefordert werden. Partielle Abhängigkeiten von Teilen des Schlüssels<sup>90</sup> können z. B. Probleme bereiten, treten aber bei der Verwendung synthetischer Schlüssel<sup>91</sup> nicht auf.

Die Entscheidung für eine objektorientierte Datenbank würde jedoch das Programm längerfristig an Java binden,<sup>92</sup> was im Hinblick auf den kürzlich vollzogenen Wechsel von C++ zu Java nicht akzeptabel bezüglich der Zukunftssicherheit erscheint. Ein vergleichbares Problem ist beim relationalen Konzept nicht zu erwarten, da es von den meisten Programmiersprachen unterstützt wird.

Durch die lange Historie ihrer Entwicklung und wegen ihrer starken Verbreitung haben relationale Systeme auch Reifegrade, insbesondere in Bezug auf Transaktionssi-

---

<sup>88</sup> Diese betrifft primär doppelte Spalten für identische Attribute in unterschiedlichen Tabellen. Jedoch werden keinerlei Daten doppelt gespeichert, da jedes Objekt nur in einer Tabelle gespeichert wird.

<sup>89</sup> Abgesehen von der mehrfachen Speicherung des Schlüssels.

<sup>90</sup> Ein Schlüssel einer Relation kann aus mehreren Attributen bestehen. Lässt sich aus dem Wert eines solchen Attributs direkt auf den Wert eines anderen schließen, liegt eine partielle Abhängigkeit vor.

<sup>91</sup> Gibt es für ein Objekt keinen natürlichen Schlüssel (z. B. der Ländername), muss ein synthetischer erzeugt werden. Dieser ist meistens eine fortlaufende Ganzzahl und besteht aus nur einem Attribut, sodass keine partiellen Abhängigkeiten mehr auftreten können.

<sup>92</sup> Die Möglichkeit zum Datenexport und -import in eine andere Datenbank besteht natürlich. Ein verlustfreier Übertrag ist jedoch nicht ohne Weiteres gewährleistet.

cherheit und des Mehrbenutzerbetriebs erreicht, die objektorientierte Lösungen<sup>93</sup> zum Teil noch vermissen lassen. Zudem werden auch viele kommerzielle und sehr leistungsfähige Systeme angeboten, sodass bei entsprechender Ausgestaltung der Schnittstellen ein späterer Wechsel zu einer leistungsfähigeren Variante bei Bedarf ohne Weiteres möglich erscheint.

Des Weiteren ist davon auszugehen, dass beim objektorientierten Ansatz die Neigung, das bestehende Datenformat zu optimieren, sehr viel geringer ist, da keine Probleme bei der Speicherung zu erwarten sind. Hierdurch werden schlechte Implementierungen tendenziell länger „überleben“.

Um die Abbildung der Objekte auf Relationen zu vereinfachen und zu vereinheitlichen, gibt es entsprechende Frameworks (Konzepte und Bibliotheken). Zum Zeitpunkt der Entscheidung war der neue EJB (Enterprise Java Beans) 3.0 Standard noch nicht verfügbar, sodass Hibernate und Hibernate Annotations als objektrelationales Mapping<sup>94</sup> (ORM) Werkzeug benutzt wird. Dieser Ansatz bietet den Vorteil, dass die Elemente, die explizite Mapping-Anweisungen erfordern, durch Annotationen direkt im Quellcode sichtbar sind und somit auch weiterhin nur Plain Old Java Objects (POJO) zum Einsatz kommen können. Ein Großteil der Spezifikationen der Hibernate-Annotations hat auch mittlerweile Einzug in die Spezifikation von EJB 3.0<sup>95</sup> erhalten, sodass ein Wechsel mit relativ geringen Änderungen möglich sein sollte. Die neue EJB- Spezifikation verzichtet nun auch im Gegensatz zur Version 2.0 auf die Remote- und Homeinterfaces.<sup>96</sup>

## 4.7 Weboberfläche

Im Ergebnis soll eine unabhängige Lösung implementiert werden, um es Anwendern zu ermöglichen, den Inhalt der Datenbank einzusehen, ohne die Desktop-Anwendung nutzen und verstehen zu müssen. Diese sollte natürlich nicht nur einfach die in Tabellen hinterlegten Daten ausgeben, sondern diese benutzerfreundlich aufbereiten.

---

<sup>93</sup> „Da objektorientierte Datenbanksysteme eine verhältnismäßig neue Technologie darstellen, hat sich bislang noch kein standardisiertes Objektmodell oder eine Anfragesprache so durchgesetzt, wie es bei SQL im relationalen Bereich der Fall ist. Die Vielfalt hat bisher die Portabilität von objektorientierten Datenbankanwendungen stark eingeschränkt.“ Kemper, A./Eickler, A., 2009, S. 376.

<sup>94</sup> Vgl. Berger, R. F., et al., Hibernate, 2006, S. 9.

<sup>95</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 301.

<sup>96</sup> Vgl. Wutka, M., J2EE guide, 2002, S. 201.

Um einen möglichst einfachen Zugang zu den Informationen des Archivs zu gewährleisten, erscheint eine webbasierte Lösung am sinnvollsten. Ein Browser ist i. d. R. bereits auf jedem PC vorhanden und erfordert somit keine separate Installation. Zudem ist sichergestellt, dass immer die aktuellste Version benutzt wird, da diese auf dem Server zentral gepflegt wird.

Aufgrund der klaren Trennung von Dokumentations- und Rechercheaufgaben von der eigentlichen Berechnung sowie der erhöhten Flexibilität gegenüber kommenden Anforderungen, wird diesem Konzept der Vorzug gegenüber einer in einer Desktopversion integrierten Lösung gegeben.

Das Webinterface soll primär vier Aufgaben erfüllen:

- Die Anwendung soll die Details eines jeden Landes darstellen können. Die Darstellung soll an die Eingabemasken der Desktop-Version angelehnt werden, da sich diese bereits als vorteilhaft erwiesen haben.
- Die Anzeige der Dokumentation (insbesondere der PDF-Dateien) ist zu ermöglichen. Der Upload solcher Dokumente soll auch unterstützt werden. Sonstige Modifikationen, z. B. an Textkommentaren sind nicht erwünscht, da diese i. d. R. spezifische Informationen bezüglich der konkreten Modellierungen enthalten, sodass Änderungen über die Desktop-Version erfolgen sollen.
- Einzelne Details sind durch gezielte Abfragen zu erheben. Die Steuersätze einzelner Steuern sollen z. B. für verschiedene Jahre oder Länder ermittelt werden können. Zur Vermeidung häufiger Änderungen an der Webschnittstelle sollen Anfragen (z. B. über SQL) im Webinterface erstellt und verwaltet werden können. Zudem sollen alle Nutzer bereits erstellte Anfragen ausführen können.
- Die Recherche nach Steuerarten und Dokumentationen wird ermöglicht. Durch eine Klassifizierung der Steuerarten soll die Suche nach einzelnen Steuerarten und der dazugehörigen Dokumentation vereinfacht werden.

#### 4.8 Refaktorisierung & Reverse Engineering oder Neuentwurf

Unter Refaktorisierung versteht man den Veränderungsprozess eines Softwaresystems dergestalt, dass die interne Struktur des Codes verbessert wird, sich das Verhalten nach außen hin aber nicht verändert.<sup>97</sup> Mit vertretbarem Aufwand ist dies i. d. R. jedoch nur begrenzt möglich, bzw. der Aufwand für einen Neuentwurf ist geringer.<sup>98</sup> Unter allen Einflussfaktoren abzuwägen, ist keine triviale Aufgabe.

Sollte der Weg der Optimierung des bereits bestehenden Programmcodes beschritten werden, bietet es sich an, daraus das Datenbankschema (Reverse Engineering)<sup>99</sup> abzuleiten. Dieser umgekehrte Entwurf erzeugt jedoch häufig ein suboptimales Schema, sodass auch bei der Refaktorisierung die Implikationen für die Abbildung der Daten zu berücksichtigen sind.

Zum Zeitpunkt der Übernahme des Projektes wurde gerade das erste Mal mit der neuen Version des European Tax Analyzer, die intern<sup>100</sup> als Variable International Tax Analyzer<sup>101</sup> bezeichnet wird, gerechnet. Zudem gestaltete sich die Schätzung des benötigten Aufwands aufgrund der Komplexität, der Detailtiefe und der funktionalen Abhängigkeiten einzelner Module als sehr schwer. Daher wurde der Refaktorisierung zunächst der Vorzug gegeben.

Im Laufe der Entwicklung wurde aber immer deutlicher, dass zumindest gewisse Teilmodule neu geschrieben werden mussten. Auch eine zwischenzeitlich verfügbare neue Sprachversion<sup>102</sup> eröffnete neue Möglichkeiten. Dies führte dazu, dass es notwendig war, einen neuen Entwicklungszweig anzulegen.

Dies birgt jedoch das Risiko, dass die Zweige „zu weit auseinander treiben“.<sup>103</sup> Dies bedeutet, dass eine potentielle Zusammenführung zu viele Konflikte verursachen würde und somit kaum noch möglich ist. Dieser Umstand ließ sich jedoch aufgrund der gravierenden Änderungen nicht vermeiden.

---

<sup>97</sup> Vgl. Fowler, M., Refactoring, 1999, S. XVI.

<sup>98</sup> Vgl. Fowler, M., Refactoring, 1999, S. 66.

<sup>99</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 35.

<sup>100</sup> Zwar wurde dieser Name von Stetter eingeführt, wurde aber zugunsten des bekannteren Namens nicht offiziell verwendet. Vgl. Stetter, T., VITAX, 2005, S. 3.

<sup>101</sup> Stetter, T., VITAX, 2005, S. 3.

<sup>102</sup> Gemeint ist hier der Wechsel von Java 1.4 zu Java 5 (1.5).

<sup>103</sup> Collins-Sussman, B., et al., Subversion, 2008, S. 93.

Es bleibt somit festzuhalten, dass der bereits vorhandene Quellcode zu optimieren und zum Teil neu zu entwerfen ist. Bei diesen Änderungen sind die Implikationen auf das Datenbankschema zu berücksichtigen und diese zu optimieren. Das Schema wird implizit erzeugt, indem dem Quellcode durch Annotationen entsprechende Abbildungsanweisungen hinzugefügt werden. Hierdurch verschmilzt der Entwurf der Datenbank und des Programmcodes zu einem Arbeitsschritt.

#### **4.9 Data Access Objects (DAO)**

“Ein DAO kapselt den Zugriff auf eine Datenquelle und die verwendete Abfragesprache. Es stellt der Anwendung eine definierte Schnittstelle bereit, mit der der Zugriff auf die Daten erfolgt.“<sup>104</sup> Diese dienen somit der Kapselung der Datenbankzugriffe und entsprechen der Datenhaltungsschicht in der klassischen Dreischichtenarchitektur<sup>105</sup> (Präsentations-, Geschäftslogik- und Datenebene).

Normalerweise wird für jedes Datenobjekt und jede unterstützte Datenbank ein DAO angelegt. Diese Objekte enthalten die datenbankspezifischen Abfragen zum Laden und Speichern der einzelnen Attribute der Datenobjekte. Hierbei wird normalerweise dem Konstruktor die zugehörige Datenbankverbindung übergeben. Durch die uniforme Ausgestaltung (i. d. R. durch Interfaces) muss nur die konkrete Instanz ersetzt werden und datenbankspezifische Details bleiben den restlichen Programmteilen verborgen.

DAOs sind aufgrund der gewählten Infrastruktur jedoch nicht notwendig, da einerseits der zu verwendende Datenbankdialekt ein Konfigurationsdetail von Hibernate darstellt und somit der Bedarf an verschiedenen DAOs für verschiedene Datenbankmanagementsysteme (DBMS) entfällt.<sup>106</sup> Andererseits sind, durch die Verwendung von Annotationen, nur noch selten direkte Abfragen im Quellcode notwendig.

Somit wäre die explizite Verwendung von DAOs für jede Datenklasse unnötiger Overhead, sodass bei der Desktopanwendung darauf verzichtet wird. Bei der Webanwendung hingegen wird dieses Design-Pattern benutzt, da hier vermehrt explizite Anfragen zur Performanceoptimierung zur Anwendung kommen.

---

<sup>104</sup> Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 74.

<sup>105</sup> Wutka, M., J2EE guide, 2002, S. 46 f.

<sup>106</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 74.

#### 4.10 Sicherheit

Hierbei wird davon ausgegangen, dass die verwendete Infrastruktur, wie z. B. das Betriebssystem des Servers, die Datenbank als Software usw. bereits ausreichend sicher ist. Somit liegt der Fokus primär auf den Anforderungen für die zu erstellende Software. Diese hängen aber wiederum sehr stark vom beabsichtigten Einsatzgebiet und von der Einsatzumgebung ab.

Diese Anwendung wird für die Verwendung im Intranet konzipiert und geht von einem wohlwollenden Benutzer aus, sodass prinzipiell keine gesonderten Voraussetzungen bzgl. der Sicherheit erfüllt werden müssen. Trotzdem wurde versucht, bekannte Sicherheitsrisiken zu vermeiden, soweit der zusätzliche Aufwand vertretbar war.

Hierzu gehören unter anderem die Verschlüsselung der Benutzerpasswörter und die Vermeidung von SQL-Injections<sup>107</sup> (Eingabe von SQL-Code bei Benutzereingaben) durch Prepared-Statements<sup>108</sup> sowie die Validierung von Benutzereingaben bei der Weboberfläche. Zudem erscheint es zweckmäßig, alle direkten Datenbankabfragen der Rolle „Administrator“ vorzubehalten.

Des Weiteren wurde eine sichere SSL-Verbindung für das Webinterface verwendet. Ein kritisches und nicht leicht zu lösendes Problem stellt auch die Bereitstellung der Authentifizierungsdaten des Datenbankbenutzers dar, da dieser der Anwendung bekannt sein muss. Eine erste Verbesserung kann hier das Vermeiden von Konfigurationsdateien und somit die Verlagerung in den Quellcode sein. Eine weitere Verschlüsselung dieser Daten erscheint ebenso hilfreich. In Kombination mit einem starken Passwort und einem nicht leicht zu erratenden Benutzernamen sollte dies ausreichen.

Bei Änderungen der Einsatzbedingungen, wie z. B. bei einem öffentlichen Zugriff,<sup>109</sup> müsste man wohl ein eigenes Sicherheitskonzept entwickeln und dessen Umsetzung und Einhaltung sicherstellen. Hierbei müsste auch eine realistische Kosten-Nutzen-Abwägung erfolgen, da z. B. der Einsatz von Hardware gestützten Kryptografiever-

---

<sup>107</sup> Vgl. Bachfeld, D., Giftspritze, <http://www.heise.de/.../artikel/43175>.

<sup>108</sup> Prepared-Statements sind vorgefertigte Anfragen, bei denen nur noch die variablen Werte ersetzt werden.

<sup>109</sup> Hiermit ist der direkte Zugriff eines beliebigen Rechners über das Internet gemeint.

fahren<sup>110</sup> die Verhältnismäßigkeit wohl sprengen würden. Bei einer solchen Erweiterung des Einsatzrahmens müsste man sicherlich auch überlegen, ob evtl. eine Validierung des Konzeptes durch externe Dienstleister sinnvoll ist.

Generell ist es fast unmöglich, die Fehlerfreiheit einer Software zu garantieren oder zu gewährleisten. Sogar viele bereits bekannte Sicherheitslücken in populären Anwendungen werden nicht immer behoben.<sup>111</sup>

#### **4.11 Ergebnisse**

Somit bleibt festzuhalten, dass es notwendig ist, das bisherige Datenmodell zu reorganisieren und aufzuspalten. Diese Aufteilung sieht die getrennte Speicherung der Daten des ökonomischen Modells und der steuerlichen Daten vor. Die Verwendung einer relationalen Datenbank wird zur Speicherung der Modellierungen der steuerlichen Sachverhalte vorgeschlagen.

Für das Unternehmensmodell erscheint es zunächst zweckmäßig, es weiterhin durch eine neu entworfene Version der XML-Serialisierung zu realisieren. Dies erscheint vor allem deswegen sinnvoll, weil diese zurzeit vermeidet, dass es zu einer unbeabsichtigten Vermischung der Datenmodelle kommt. Des Weiteren wird auch für den Anwender die Trennung klarer. Zudem ist bei den Unternehmensmodellen i. d. R. keine Evaluationsgeschichte zu beobachten, sodass es sich normalerweise um isolierte Instanzen handelt. Somit ist eine Versionshistorie und Benutzerverwaltung nicht zwingend erforderlich. Trotzdem stellt eine spätere Umstellung auf eine weitere Datenbank eine zukünftige Option dar.

Nachdem der Programmcode entsprechend optimiert wurde, wird mithilfe entsprechender Abbildungsvorschriften anschließend ein Datenbankschema erzeugt. Bei dieser Optimierung muss darauf geachtet werden, dass Vererbungshierarchien und Beziehungen möglichst effizient abgebildet werden, um mögliche Performanzprobleme auszuschließen.

Des Weiteren soll zur Unterstützung des Mehrbenutzerbetriebs die optimistische Sperrstrategie umgesetzt werden. Zur Unterstützung der Dokumentation der Steuer-

---

<sup>110</sup> Vgl. IBM, Cryptographic Coprocessor, <http://www-03.ibm.com/.../overproduct.shtml>, 09. 02. 2009.

<sup>111</sup> Vgl. Bachfeld, D., Sicherheitslücken bleiben ungepatcht, <http://www.heise.de/.../126785>.

daten wird die Verwendung des PDF-Formats vorgeschlagen. Um eine möglichst flexible und leicht zu bedienende Recherche-Möglichkeit zu erhalten, wird die Entwicklung einer Weboberfläche angestrebt. Nur für diese erscheint die Verwendung von Data Access Objects sinnvoll.



## 5 Umsetzung der Anforderungen mithilfe ausgesuchter Methoden

Dieses Kapitel behandelt primär die Implementierungsdetails der konkreten Umsetzung und nimmt Abwägungen bzgl. verschiedener Optionen vor. Zunächst werden prinzipielle Entscheidungen hinsichtlich der persistenten Datenspeicherung diskutiert und im Hinblick auf das zu erwartende Benutzerverhalten evaluiert.

Der zweite Teil erläutert detailliert, welche Änderungen und Erweiterungen am bisherigen Datenmodell vorgenommen wurden, um ein möglichst dauerhaftes Schema zu erhalten. Zunächst wird das Ländermodell beschrieben. Danach wird die darin enthaltene Komponente Unternehmensmodell in ökonomische, steuerliche Wahlrechte, wie z. B. Abschreibungen, und steuerliche Daten zerlegt. Diese Wahlrechte sind im European Tax Analyzer kein direkter Bestandteil der Bemessungsgrundlagen, da sie bereits zur Ermittlung des Gewinns benötigt werden. Auf dieses Resultat kann über die Bemessungsgrundlage (BMG) der Gewinn- und Verlustrechnung zugegriffen werden und zur Berechnung der Steuern verwendet werden. Im Anschluss daran werden die Steuerarten und ihre Bemessungsgrundlagen beleuchtet.

### 5.1 Transaktionsverwaltung

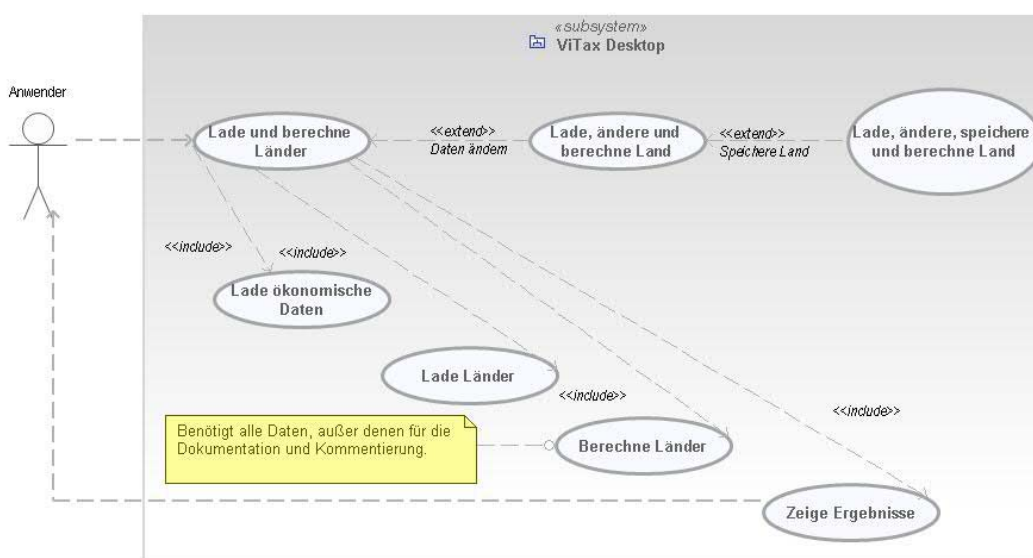
Die Datenbank stellt das Konzept der Transaktionen zwar bereit und garantiert die Einhaltung des ACID-Paradigma (vgl. 3.1); die Transaktionen müssen allerdings durch das Programm erstellt und verwaltet werden. Zudem besteht bei einer Desktopanwendung das Problem, dass die Zeitspanne zwischen dem Lesen der Daten und ihrer Speicherung sehr groß werden kann. Dies gefährdet insbesondere die Anforderungen bzgl. der Isolation und kann zu einer schlechten Performanz führen, da das Datenbankmanagementsystem evtl. viele Transaktionen sowie die dazu gehörigen Daten (z. B. Sperren) über einen langen Zeitraum verwalten muss. Zudem sinkt die Wahrscheinlichkeit des erfolgreichen Abschlusses, sodass viele Aufträge erneut gestartet werden müssen.

Durch die Wahl der optimistischen Sperrstrategie in Kombination mit dem *Session-Per-Request-Pattern* (eine Sitzung bzw. Transaktion pro Anfrage) entfallen diese Probleme, da die notwendigen Sperren erst beim Versuch, Änderungen zu persistieren, angefordert werden. Der reine Ladevorgang verursacht keine Sperren.

### 5.1.1 Lange versus kurze Konversationen

Generell ist auch das Anforderungsprofil von „Thick-Client Applications“<sup>112</sup> ein ganz anderes als von Webanwendungen. Im konkreten Fall wird bei der Desktop-Anwendung fast immer eine Berechnung (vgl. Abbildung 11) durchgeführt. Bei dieser werden aber alle Daten, mit Ausnahme der Dokumentation, benötigt. Somit macht es keinen signifikanten Unterschied, ob diese Werte bereits beim Laden des Landes oder erst bei der eigentlichen Kalkulation aus der Datenbank gelesen werden.

Abbildung 11: Standard Use Cases



Standardmäßig lädt *Hibernate* alle 1:n und m:n Beziehungen dynamisch bei Bedarf nach. Dies hat den Vorteil, dass aufwendige Datenbankabfragen vermieden werden. Hierbei wird jedoch eine dauerhaft geöffnete Sitzung vorausgesetzt. Aufgrund der Konvention, dass bei einer Ausnahme eine Session ungültig wird und geschlossen werden muss, ist diese Anforderung jedoch kaum erfüllbar. Dies wird zudem dadurch erschwert, dass es keinen definierten Einstiegspunkt gibt, um solche dynamischen Nachladeversuche abzufangen oder eine adäquate Session zu gewährleisten. Dies stellt insbesondere deshalb ein Problem dar, weil die Sitzung nicht zustandslos ist. Dieser Verbindung muss das Objekt, das die Nachladeanforderung auslöst, be-

<sup>112</sup> In diesem Fall sind alle nicht webbasierten Anwendungen gemeint, insbesondere Desktop-Anwendungen, die größere Datenmengen verarbeiten. Shadow, G., Thick-Client Applications, <https://www.hibernate.org/333.html>, 08. 09. 2009.

reits bekannt sein. Somit reicht es nicht aus, jede geschlossene Sitzung einfach durch eine neue zu ersetzen.

Daher wurde auf die Möglichkeit des dynamischen Nachladens weitestgehend verzichtet und das *Session-Per-Request-Pattern*<sup>113</sup> implementiert, das die Sitzung nach jeder Anfrage schließt. Dies hat den Vorteil, dass ungültige Sessions fast nicht mehr vorkommen. Zudem ist so auch die Fehleranalyse einfacher, da die Requests in der Regel nur eine kurze Transaktion beinhalten und sich nicht gegenseitig beeinflussen können.

Somit bleibt festzuhalten, dass es aufgrund der Anforderungen der Use-Cases und von Problemen beim dynamischen Nachladen sinnvoll erscheint, alle benötigten Daten auf einmal zu laden. Daher erscheint es auch nicht zweckmäßig, die GUI-Dialoge aufwendig zu refaktorisieren, um jeweils nur die benötigten Teile des Gesamtmodells zu laden. Deswegen wird nun immer das komplette Land, mit Ausnahme der PDF-Dokumente, geladen. Diese Dateien werden aufgrund ihrer Größe erst bei Bedarf nachgeladen. Dies stellt insofern kein Problem dar, da die Nachladeanforderung in den Zugriffsmethoden erfolgen kann und hierdurch die oben genannten Probleme umgangen bzw. behandelt werden können. Die Realisation erfolgte über die Markierung der Beziehungen als „*eager*“ (dt. eifrig – immer laden). Die Beziehungen zu den PDF-Dokumenten wurden mit „*lazy*“ (dt. faul – erst bei Bedarf nachladen) annotiert.

Der gravierende Nachteil besteht jedoch darin, dass bei den Standardparametern die gesamte Datenbank via *Left-Outer-Joins*<sup>114</sup> geladen wird, was zu massiven Speicher- und Performanceproblemen sowie zu einer hohen Netzlast führt. Dies wurde dadurch gelöst, dass Ergebnisse vorheriger Abfragen für gezielte Projektionen benutzt<sup>115</sup> und somit Joins weitestgehend vermieden werden. Des Weiteren wurden 1:n-Beziehungen auf der N-Seite gemappt und somit zusätzliche Join-Tabellen umgangen. 1:1-Verbindungen von nicht Standardattributen<sup>116</sup> wurden mit *@Fetch(value = FetchType.SELECT)* annotiert. Die mit diesen Parametern automatisch durch Hi-

---

<sup>113</sup> Vgl. O. V., Hibernate Reference Manual, S. 125 f.

<sup>114</sup> Zur Erklärung von *Left-Outer-Joins* siehe Kemper, A./Eickler, A., 2009, S. 81.

<sup>115</sup> Dies wurde primär mithilfe des Parameters *org.hibernate.cfg.Environment.MAX\_FETCH\_DEPTH* mit dem Wert von Null erreicht. Vgl. O. V., Hibernate Reference Manual, S. 34.

<sup>116</sup> Diese werden implizit mit *@Basic* annotiert.

*bernate* erzeugte Abfrage<sup>117</sup> ist zwar deutlich länger als zuvor, verhindert aber die obigen Probleme.

Bei der Webanwendung ist dies nicht der Fall. Sie dient primär Recherchezwecken und benötigt daher keinen kompletten Datensatz eines Landes. Während einer Sitzung kann es natürlich vorkommen, dass alle Parameter abgerufen werden. Dies sollte aber äußerst selten der Fall sein, da für einzelne Seiten nur kleine Teilmengen der Daten eines Landes benötigt werden. Somit müssten quasi alle Seiten, die zu einem Land gehören, sequentiell aufgerufen werden, um den kompletten Datensatz zu lesen.

Zudem ist der Workflow des Benutzers nicht vorhersagbar, sodass nur die aktuell anzuzeigende Seite relevant ist. Somit erscheint es sinnvoll, immer nur die benötigten Daten gezielt zu laden. Im Gegensatz zum „*Open-Session-In-View*“<sup>118</sup>-Ansatz, ergibt sich hierbei der Vorteil, dass das *Session-Per-Request-Pattern* zur Anwendung kommen kann. Hierdurch werden Fehler schneller entdeckt, da bei nicht geladenen Daten eine Ausnahme erzeugt wird. Ansonsten würden diese einfach stillschweigend nachgeladen werden.

### 5.1.2 Sperrstrategie

Durch die Wahl dieses Patterns scheiden lange Konversationen und somit auch lange Transaktionen aus. Dies hat zur Folge, dass auch pessimistische Sperren nicht mehr sinnvoll anwendbar sind. Das Aufrechterhalten über das Transaktionsende hinaus ist nicht möglich und sinnvoll, da ansonsten nicht garantiert werden kann, dass diese Verbote auch wieder aufgehoben werden. Zudem würde dies gegen die Anforderung der Isolation verstoßen, da beide Transaktionen direkt voneinander abhängen würden. Bei Verwendung dieser expliziten Zugriffsbeschränkungen und gleichzeitiger kurzer Konversation müsste sonst zu Beginn jeder schreibenden Transaktion geprüft werden, ob die geladenen Daten noch aktuell sind oder nicht. Dies garantiert aber schon die Transaktion an sich durch die benötigte Isolation, sodass keine expliziten

---

<sup>117</sup> Bei der Benutzung von HQL oder Criteria-Queries erzeugt Hibernate automatisch eine entsprechende SQL-Abfrage. Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 103.

<sup>118</sup> Die Datenbankverbindung bleibt zumindest bis zur Darstellung der Webseite geöffnet. Vgl. Red Hat Middleware, <https://www.hibernate.org/43.html>, 08. 09. 2009; Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 42 und 96 ff.

Sperren mehr notwendig sind. Der einzige Unterschied liegt im Zeitpunkt des Scheiterns. Beim optimistischen Verfahren wird die Transaktion wahrscheinlich erst im weiteren Verlauf abgebrochen, während beim pessimistischen diese tendenziell<sup>119</sup> eher zu Beginn scheitert. Generell ist bei einer Desktopanwendung die Verwendung einer solchen Strategie problematisch. Im Falle einer nicht abgeschlossenen langen Transaktion könnte das gesamte System durch Arbeitsunterbrechungen (z. B. wenn der Anwender in die Mittagspause geht oder Feierabend macht), aufgrund von nicht aufgehobenen Verboten, blockiert werden. Des Weiteren ist die pessimistische Version von Sperren auch nur für wenige gleichzeitige Benutzer geeignet, da sich sonst Kollisionen nur schwer vermeiden lassen.

Somit wurde das optimistische Verfahren ausgewählt. Hierbei wird bei jedem Objekt, das getrennt gespeichert werden kann, ein Versionsattribut hinzugefügt. *Hibernate* lehnt dann automatisch Änderungen ab, wenn dieses Attribut in der Zwischenzeit erhöht wurde. Dies wird dem Anwender dann mitgeteilt.<sup>120</sup>

Dadurch kann es jedoch vorkommen, dass der Anwender Änderungen erneut vornehmen muss und so eine Art von Datenverlust entsteht, da er die geänderten Werte erst wieder laden muss. Dies kann bei der pessimistischen Strategie nicht passieren, da zwischenzeitliche Änderungen ausgeschlossen sind. Andernfalls würde schon der vorhergehende Versuch, ein Datum zu sperren, scheitern.

Durch die Möglichkeit der temporären Zwischenspeicherung tritt aber kein tatsächlicher Verlust ein. Der Anwender muss, nach Abgleich der vorherigen Revision, seine Überarbeitungen aber evtl. erneut eingeben.

Beim derzeitigen und dem zu erwartenden Benutzerverhalten erscheint dieser Nachteil akzeptabel, da ein gleichzeitiges, konkurrierendes Bearbeiten der gleichen Datensätze eher unwahrscheinlich ist. Zurzeit gibt es relativ wenige Benutzer und recht viele verschiedene Datensätze, sodass Kollisionen eher selten auftreten dürften. Des Weiteren ist dies aufgrund der Arbeitsteilung in den einzelnen Projekten und Ab-

---

<sup>119</sup> Der genaue Zeitpunkt hängt aber von der jeweiligen Historie ab. Sollte ein Datum erst recht spät explizit gesperrt werden, könnte der Zeitpunkt nur minimal von der optimistischen Strategie abweichen.

<sup>120</sup> Es wird eine Ausnahme ausgeworfen, die die Anwendung entsprechend behandelt und für den Benutzer aufbereitet.

sprachen untereinander nahezu auszuschließen. Sollte dieser Aspekt in Zukunft ein Problem darstellen, kann dem durch Erhöhung der Granularität entgegen gesteuert werden. Um bei Bedarf die spätere separate Speicherung von Steuerobjekten zu ermöglichen, wurde diese bereits mit einem Versionsattribut versehen, das aber bisher nicht benutzt wird. Hierdurch würden dann Versionsinformationen über das Land verloren gehen, da eine Änderung an einer Steuer ja auch zugleich eine Änderung am Ländermodell bedeutet. Andernfalls würde die notwendige, zusätzliche Aktualisierung der Bearbeitungshistorie die erhöhte Granularität konterkarieren, da nun wieder das ursprüngliche Land geändert werden müsste.

## 5.2 Kopierinfrastruktur

Bisher wurde ein Land dadurch kopiert, dass man es als Datei gespeichert und diese erneut geladen hat. Dies ist notwendig, um verschiedene Variationen mit möglichst geringem Aufwand zu erzeugen. Will man z. B. eine Reformoption prüfen, wird man am ehesten auf einem Duplikat der aktuellen Rechtslage aufbauen.

Angesichts der synthetischen IDs (vgl. 5.2.2) vieler Objekte und der inkompatiblen Klassen<sup>121</sup> scheidet diese Vorgehensweise aus. Daher musste dafür nun eine programmatische Lösung etabliert werden. Aufgrund von Problemen mit der *clone*-Infrastruktur<sup>122</sup> (z. B. mit finalen Attributen), die eigentlich für diese Aufgabe gedacht war, wurde eine eigene Lösung entwickelt.

Diese besteht primär darin, dass ein einfaches Interface (*MixinCloneable*) deklariert wurde. Zudem wurde die Anforderung postuliert, dass jede Klasse, die dieses Interface implementiert, einen Kopierkonstruktor bereitstellt und diesen in der implementierten Methode aufruft. Dadurch ist zum einen die Wahrung der Polymorphie mithilfe des Überschreibens der Methode gewährleistet, zum anderen sind auch die Nachteile von *clone* bzgl. finaler Attribute beseitigt.

Im Normalfall werden Objekte nicht mehrfach referenziert, sodass der direkte Aufruf der Kopierroutine beim referenzierenden Objekt erfolgen kann. Bei multiplen Referenzen stellt sich das Problem, dass beim Aufruf der Kopierfunktion nicht klar ist, ob

---

<sup>121</sup> *Hibernate* verwendet eine eigene Implementierung von Collection-Klassen mit denen der *XMLEncoder* nicht umgehen kann.

<sup>122</sup> Vgl. Bloch, J., *Effective Java*, 2008.

es bereits eine Kopie dieses Objektes gibt. Existiert diese bereits, darf kein weiteres Duplum erstellt werden. Die Referenz muss dann so angepasst werden, dass sie auf die bereits angelegte Kopie verweist. Daher war es besonders kompliziert, die Referenzen auf andere Steuern adäquat zu ersetzen, da diese durch das Erzeugen einer Kopie ungültig werden. Nach Abschluss des Duplizierens der Steuersysteme müssen diese nach veralteten Verweisen durchsucht und durch Verknüpfungen mit den Duplikaten ersetzt werden. Dies betrifft auch die Herstellungskosten, falls Steuern einbezogen waren und die Steuersysteme der Anteilseigner. Bei anderen Objekten tritt dieses Problem nicht auf, da entweder nur auf Substitute<sup>123</sup> verwiesen wird oder keine multiplen Referenzen auftreten.

Würde immer das Ziel der Referenz (z. B. „*OtherTaxesElement*“ -- vgl. 5.3.7.2.2) kopiert, würden in diesem Fall multiple Dubletten angefertigt. Somit würde keinerlei Verbindung mehr zum Steuerobjekt aus dem eigentlichen Steuersystem bestehen. Diese Reproduktion würde dann unter Umständen nicht einmal berechnet werden. Spätestens aber bei Modifikationen an der Steuer würde sich offenbaren, dass es sich um zwei getrennte Ausprägungen handelt. Somit muss dieser Verweis nachträglich korrigiert werden, nachdem das Duplum vom ursprünglichen Objekt erzeugt worden ist.

### 5.2.1 XML-Encoder

Aufgrund der oben schon erwähnten Inkompatibilität der *Collection*-Klassen muss bei jedem Land, das aus der Datenbank geladen wurde, vor der temporären, lokalen Speicherung eine Kopie erstellt werden. Erst diese Kopie kann dann entsprechend gespeichert werden, da diese dann keine inkompatiblen Klassen<sup>124</sup> mehr enthält.

Des Weiteren konnten auch keine EnumMaps benutzt werden, da diese keinen parameterlosen Standardkonstruktor anbieten. Benötigte Attribute für andere Konstrukto-  
ren sind nicht öffentlich zugreifbar, weshalb auch die Implementierung eines *Persis-*

---

<sup>123</sup> In den Herstellungskosten wird z. B. die Maschine eins über den entsprechenden Aufzählungstyp referenziert. Vgl. 5.3.3 und Abbildung 15.

<sup>124</sup> Diese wurden während der Kopie ersetzt.

*tenceDelegate* unmöglich ist. Daher wurde in solchen Fällen eine *TreeMap*<sup>125</sup> verwendet.

### 5.2.2 Datenbank: ID und Versionsattribute

Beim Erstellen einer Kopie würde die Datenbank es prinzipiell ablehnen, ein Objekt mit der gleichen ID erneut zu speichern, da hierdurch die Bedingungen für ein Schlüsselattribut verletzt würden. Somit müsste man vor einer Speicherung dieses Attribut<sup>126</sup> ändern. Jedoch muss auch das Versionsattribut wieder auf *null* gesetzt werden, damit es einer neuen Instanz entspricht. Dies erscheint auf den ersten Blick recht einfach, wird im konkreten Fall aber sehr schwierig, da ein Land aus einer komplexen, baumähnlichen Datenstruktur mit sehr vielen verschiedenen Instanzen besteht. Somit müsste auch für jede Klasse eine Methode implementiert werden, die dann für alle Beziehungen den Aufruf zum Reset der datenbankspezifischen Attribute weitergibt. Dies würde im Prinzip einer reduzierten Kopieroutine entsprechen und einen ähnlich hohen Aufwand erfordern. Aus diesem Grund ist die Entscheidung zugunsten der flexibleren Lösung gefallen.

Zudem behebt dieser Ansatz auch das Problem der Inkompatibilität zwischen *Hibernate* und *XMLEncoder*. Deswegen wurde auch die in der Zwischenzeit implementierte Lösung, via *Reflection-API*<sup>127</sup> die Datenstruktur zu durchlaufen und die einzelnen Instanzen zu modifizieren, verworfen. Zudem erschien diese Variante als zu fehleranfällig, da nicht hundertprozentig gewährleistet werden konnte, dass alle Spezialfälle abgedeckt werden. Das nun gewählte Verfahren löst beide Probleme.

## 5.3 Modifikationen am Datenmodell

### 5.3.1 Überblick

Die notwendigen Änderungen am Datenmodell werden nachfolgend dargelegt. Die Darstellung dieser Implementierungen enthält bereits den eigentlichen Entwurf des

---

<sup>125</sup> Dadurch dürfte die Performanz geringfügig schlechter sein, da nun ein Rot-Schwarz-Baum anstelle eines Arrays verwendet wird. Vgl. <http://java.sun.com/javase/6/docs/api/java/util/TreeMap.html> und <http://java.sun.com/javase/6/docs/api/java/util/EnumMap.html>.

<sup>126</sup> Bei Hüllklassen wie *Long* wird dafür *null* verwendet, dies entspricht dem Wert einer neuen Instanz.

<sup>127</sup> Java ermöglicht es dynamisch die Klassenstruktur zu erfragen und Instanzen zu manipulieren. Vgl. *java.lang.Class* und *java.lang.reflect.\**.



Datenbankschemas und wird daher nicht explizit erläutert. Am Beispiel von Quellcodefragmenten des *UnternehmensmodellVst* (vgl. 5.3.3) wird kurz das Verfahren beschrieben.

```
@Entity

@Inheritance(strategy = InheritanceType.SINGLE_TABLE)
@DiscriminatorColumn(name = "Pre_Posttax", discriminatorType =
    DiscriminatorType.STRING)
@DiscriminatorValue(value = "PreTax")

public class UnternehmensmodellVst implements
    RequiresResetOfCalculationData, SupportsDatabaseId,
    MixInCloneable<UnternehmensmodellVst> {
```

Mit `@Entity` wird deklariert, dass diese Klasse in der Datenbank zu speichern ist.<sup>128</sup>

`@Inheritance` legt die Vererbungsstrategie fest. In diesem Fall soll für *UnternehmensmodellVst* und *UnternehmensmodellNst* nur eine gemeinsame Tabelle (*SINGLE\_TABLE*) verwendet werden. Dies ist sinnvoll, da nur *UnternehmensmodellNst* Teil der Datenbank sein soll und somit keine ungenutzten Attribute auftreten. Bei dieser Methode wird jedoch eine Spalte zur Unterscheidung (`@DiscriminatorColumn`) dieser zwei Ausprägungen benötigt und als Wert für diese Variante *"PreTax"* (`@DiscriminatorValue`) festgelegt. Alternativ wäre auch die Strategie einer Tabelle je konkreter Klasse möglich gewesen. Dadurch würden aber zwei Tabellen erzeugt, wovon eine ungenutzt bliebe. Somit erscheint die einzelne Tabelle mit der zusätzlichen Unterscheidungsspalte sinnvoller.

```
@Id
@GeneratedValue(strategy = GenerationType.AUTO)

private Long id;
```

Mit `@Id` wird das Schlüsselattribut definiert. In diesem Fall handelt es sich um eine synthetische ID, sodass noch eine Strategie zum Erzeugen dieser großen Ganzzahl (*Long*) spezifiziert werden muss.

```
@OneToMany(fetch = FetchType.EAGER, cascade =
    javax.persistence.CascadeType.ALL)
@Cascade(value =
    org.hibernate.annotations.CascadeType.DELETE_ORPHAN)
@Fetch(value = FetchType.SELECT)

private Map<String, BewertungBilanzObjekt>
    evaluationOfCurrentValue;
```

---

<sup>128</sup> Zusätzlich muss die Klasse noch bei der Konfiguration von *Hibernate* eingetragen werden.

Hier wird eine 1:m-Beziehung definiert. Mit *FetchType.EAGER* wird das sofortige nachladen angestoßen und mit *CascadeType.ALL* wird definiert, dass bei allen Datenbankoperationen, die *UnternehmensmodellVst* betreffen, diese Beziehung auch berücksichtigt werden muss. Beim Löschen werden z. B. auch alle *BewertungBilanzObjekte* (*DELETE\_ORPHAN*) gelöscht, die der gelöschten Instanz von *UnternehmensmodellVst* zugeordnet waren. Andernfalls würden entweder nur die Verknüpfungen gelöscht (*ALL* ohne *DELETE\_ORPHAN*) oder keine explizite Aktion ausgeführt. *FetchMode.SELECT* gibt an, dass kein Kreuzprodukt der Tabellen gebildet werden soll, sondern eine Unterabfrage zu verwenden ist. Dies erhöht die Performance, da nur ein geringer Teil der Kreuztabelle für das zu lieferende Resultat relevant ist.

```
@OneToMany(fetch = FetchType.EAGER, cascade =
    javax.persistence.CascadeType.ALL)
@Cascade(value =
    org.hibernate.annotations.CascadeType.DELETE_ORPHAN)
@IndexColumn(name = "AE_NR", base = 0)
@Fetch(value = FetchType.SELECT)
@JoinColumn(name = "UModell_ID")

protected AnteilseignerVst[] aeResults;
```

Die Reihenfolge der Elemente (*AnteilseignerVst[]*) bleibt durch eine zusätzliche Index-Spalte (*@IndexColumn*) gewahrt. Mit *@JoinColumn* wird festgelegt, dass diese Beziehung in der Tabelle von *AnteilseignerVst* abzubilden ist. Dadurch wird eine sonst notwendige Verknüpfungstabelle eingespart.

```
@Enumerated(value = EnumType.STRING)

private Verbrauchsfolge-Verfahren verbrauchsFolge;
```

Das *Verbrauchsfolgeverfahren* ist die konkrete Ausprägung der Aufzählung<sup>129</sup> der unterstützten verschiedenen Verfahren zu Bewertung des Langerbestandes. *EnumType.STRING* gibt an, dass die Zeichenkettenrepräsentation anstelle von Ordinalzahlen zur Abbildung in der Datenbank verwendet werden soll.

```
private String funktionAS = "AntProdAbs";
private boolean useOfAccrucalForWarranties;
```

---

<sup>129</sup> Es wird nur hinterlegt, welches Verfahren z. B. FIFO (First In First Out) benutzt werden soll. Vgl. 5.3.4.

Attribute, die gespeichert werden sollen, für die aber keine expliziten Anweisungen erforderlich sind, brauchen keine explizite Annotation. Implizit wird *@Basic* verwendet.

#### **@Lob**

```
private String kommentarAS = "Kommentar";
```

Ein Kommentar könnte die erlaubte Länge einer normalen Zeichenkette (*varchar*) übersteigen, so dass dieser als großes Objekt (*@Lob*) markiert wurde.

#### **@Transient**

```
private transient Lagerbestand lagerbestand;
```

Soll eine Referenz oder Attribut nicht persistiert werden, muss sie als *@Transient* markiert werden. Das bereits in der Variablendeklaration stehende „*transient*“<sup>130</sup> wird nicht beachtet. Aus diesen Annotationen wird dann das Datenbankschema (vgl. 5.6) erzeugt.<sup>131</sup> Diese sind zum Teil<sup>132</sup> auch in den UML-Diagrammen<sup>133</sup> (vgl. Abbildung 12) ersichtlich.

**Abbildung 12: UML-Diagramm des UnternehmensmodellVst**



<sup>130</sup> Es wird nur von der binären Speicherung (vgl. *java.io.Serializable*) verwendet und dient in diesem Fall nur zur Verdeutlichung.

<sup>131</sup> Mit Hilfe der Eigenschaft *HBM2DDL\_AUTO* kann festgelegt werden, ob das Schema erzeugt werden soll oder nicht. Für den Produktiveinsatz sollte sie nicht oder auf *validate* (überprüfen) gesetzt werden. Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 150.

<sup>132</sup> Nicht alle verwendeten Annotationen werden von der benutzten Software zum Erstellen der Diagramme unterstützt. Zudem wurde zugunsten einer besseren Übersichtlichkeit teilweise auf einige Details bewusst verzichtet.

<sup>133</sup> Die Unified Modeling Language (UML) ist eine standardisierte Sprache zur Beschreibung von Softwaresystemen. Die graphische Notation erfolgt in Form von Diagrammen.

Das Verlustverrechnungsmodul (vgl. 5.3.7.2.3) und die Betriebliche Altersversorgung<sup>134</sup> (vgl. 5.3.6) stellen Erweiterungen zum bisherigen Datenmodell dar. Alles andere wurde entsprechend refaktoriert oder neu implementiert.

### 5.3.2 Anpassungen des Ländermodells

Bisher war der Name eines Landes frei wählbar und musste nur innerhalb eines Modells (i. d. R. Basisland plus aktuell geladene Länder) eindeutig sein. Dieser Name enthielt häufig auch Informationen über das modellierte Jahr und den Rechtsstand.

Diese Freiheitsgrade sind durch die Verwendung einer Datenbank nun jedoch nicht mehr möglich und müssen stark eingeschränkt und vereinheitlicht werden. Dies ist notwendig, um daraus eine eindeutige ID für die Datenbank abzuleiten. Schließlich soll es jedes Land für jedes Jahr und mit dem für das betreffende Jahr aktuellen Rechtsstand nur genau einmal geben. Daher scheidet auch eine synthetische ID aus, da sie diese Anforderung nicht garantieren kann. Zudem müssen *equals* und *hashCode*<sup>135</sup> entsprechend überschrieben werden, um die gleiche Objektidentität zur Laufzeit des Programms zu erhalten, wie es in der Datenbank herrscht. Trotzdem ist es möglich, zwei gleiche Instanzen<sup>136</sup> zu erzeugen, da die Objektreferenzen nicht identisch sind. Dies widerspricht jedoch den Anforderungen der Datenbank und muss daher programmatisch unterbunden werden. Andernfalls würde diese Objektidentität im besten Fall von der Datenbank zurückgewiesen<sup>137</sup> werden oder im schlechtesten zu Datenverlusten und Inkonsistenzen führen. Für das Kopieren von Ländern ist dieser Dualismus jedoch temporär erwünscht. Es muss jedoch sichergestellt sein, dass die ID vor der Speicherung geändert wird und somit das Land eine noch nicht vergebene ID erhält. Ebenso muss beim Laden aus der Datenbank darauf geachtet werden, dass nicht schon ein Objekt mit der gleichen ID existiert.

Aus diesem Grund wurde eine sogenannte zusammengesetzte ID<sup>138</sup> verwendet. Diese besteht aus einem Aufzählungstyp (Enum) für den sprachenübergreifenden eindeuti-

---

<sup>134</sup> „So kann der VITAX um andere ökonomische Sachverhalte erweitert werden, wie z. B. die betriebliche Altersversorgung oder den Emissionsausstoß.“ Stetter, T., VITAX, 2005, S. 205.

<sup>135</sup> Vgl. Bloch, J., Effective Java, 2008, S. 33 ff.

<sup>136</sup> Gleichheit zumindest bzgl. der ID Attribute





<sup>137</sup> Wenn der Vergleich der Versionsattribute scheitert.

<sup>138</sup> Composite ID.

gen Ländernamen,<sup>139</sup> der Jahreszahl, der Art der Modellierung (ebenfalls Aufzählung) und einem Kommentar, der die Modellierungsart weiter einschränken soll. Die Aufzählungen werden als Zeichenkette in der Datenbank gespeichert. Somit entfällt das Risiko, dass eine Umsortierung im Quellcode Auswirkungen auf den Datenbankinhalt hat. Dies wäre der Fall, wenn man die platzsparendere Abbildung auf eine Ganzzahl wählen würde.

Um das bisherige Verhalten der ID gegenüber dem Benutzer zu gewährleisten, wird der bisherige Name (umbenannt zu: *displayname*) nur noch für Anzeigewecke benutzt und ist kein Teil der ID mehr. Zwar wird der Standardwert aus dem Ländernamen und der Jahreszahl abgeleitet. Der konkrete Inhalt ist jedoch vom Benutzer frei wählbar<sup>140</sup> und erlaubt daher keinerlei Rückschlüsse auf andere Daten des Ländermodells. Somit besteht hier keinerlei funktionale Abhängigkeit.

Abbildung 13: ID des Ländermodells

public.countrymodel		
	comment	text (2147483647)
	name	varchar (255)
	type	varchar (255)
	year	int4
	displayname	varchar (255)
	version_optlock	int8
	währungssymbol	varchar (255)
	wechselkurs	int4
	unternehmensmodell_id	int8

Das Feld *version\_optlock* dient der Versionskontrolle für die optimistische Sperrstrategie. Die *unternehmensmodell\_id* verweist entsprechend auf das Unternehmensmodell. Dort erfolgt dann die bereits erwähnte (vgl. 4.1.1) Dreiteilung der Daten.

Zudem enthält das Ländermodell auch Verweise auf die Liste der zu Dokumentationszwecken hinterlegten PDF-Dateien und der Historie der Änderungen. Diese werden jedoch in diesen Tabellen<sup>141</sup> abgebildet und sind daher in der Abbildung 13 nicht

<sup>139</sup> Dies ist wichtig, da z. B. Germany (en) und Deutschland (de) keine verschiedenen Länder darstellen sollen.

<sup>140</sup> Theoretisch könnte dieser auch einen den Schlüsselattributen widersprechenden Inhalt aufweisen. Dies sollte aber vermieden werden.

<sup>141</sup> Auch eine Abbildung durch separate Tabellen (Join-Tabellen) wäre möglich. Diese enthalten dann i. d. R. nur die Schlüssel der verbundenen Einträge. Aus den beiden Fremdschlüsseln entsteht dann der zusammengesetzte Primärschlüssel dieser Tabelle. Für sortierte Einträge gibt es in der Regel auch noch ein Positionsattribut, das dann ebenfalls Teil des Schlüssels wird. Diese Verknüpfungstabellen wurden aber aufgrund der schlechteren Performanz nicht verwendet.

enthalten. Des Weiteren sind auch die Wirtschaftsdaten (Inflationsraten und Zinssätze) enthalten, welche jedoch aufgrund der Fokussierung auf steuerlich relevante Daten nicht in der Datenbank gespeichert werden.<sup>142</sup>

### 5.3.3 Zerlegung des Unternehmensmodells

Zur Gewährleistung von reproduzierbaren Resultaten reicht es nicht aus, Zwischen- und Endergebnisse einfach nur nicht zu speichern. Vielmehr bedarf es einer einheitlichen Infrastruktur, um diese Ergebnisse vor der erneuten Berechnung gezielt löschen zu können. Andernfalls könnte es bei verschiedenen Durchläufen aufgrund von veralteten Werten zu unterschiedlichen Resultaten kommen.

Zur einheitlichen Behandlung wird daher ein Interface (vgl. Abbildung 14 - *RequiresResetOfCalculationData*) mit einer Methode eingeführt, das von jeder Klasse, die Zwischenergebnisse speichert, implementiert werden soll. Aufgrund der Tatsache, dass Objekte zur Laufzeit darauf untersucht werden können, ob sie dieses Interface implementieren, besteht die Möglichkeit, dies auch auf dynamische Datenstrukturen anzuwenden. Des Weiteren ist jede Klasse dafür zuständig, den Aufruf an die Oberklasse und alle Kompositionen weiterzugeben. Durch diese rekursive Struktur reicht es nun aus, das Zurücksetzen der Zwischenergebnisse an der Wurzel des Objektbaums auszulösen. Durch die unten angesprochene Aufteilung ist es nun auch möglich, selektiv Teile der Zwischenergebnisse zu löschen. Daher muss das Basismodell, das in der Regel als Referenz für den Fall ohne Steuern dient, nur einmalig berechnet werden. Somit kann die Berechnung optimiert werden.

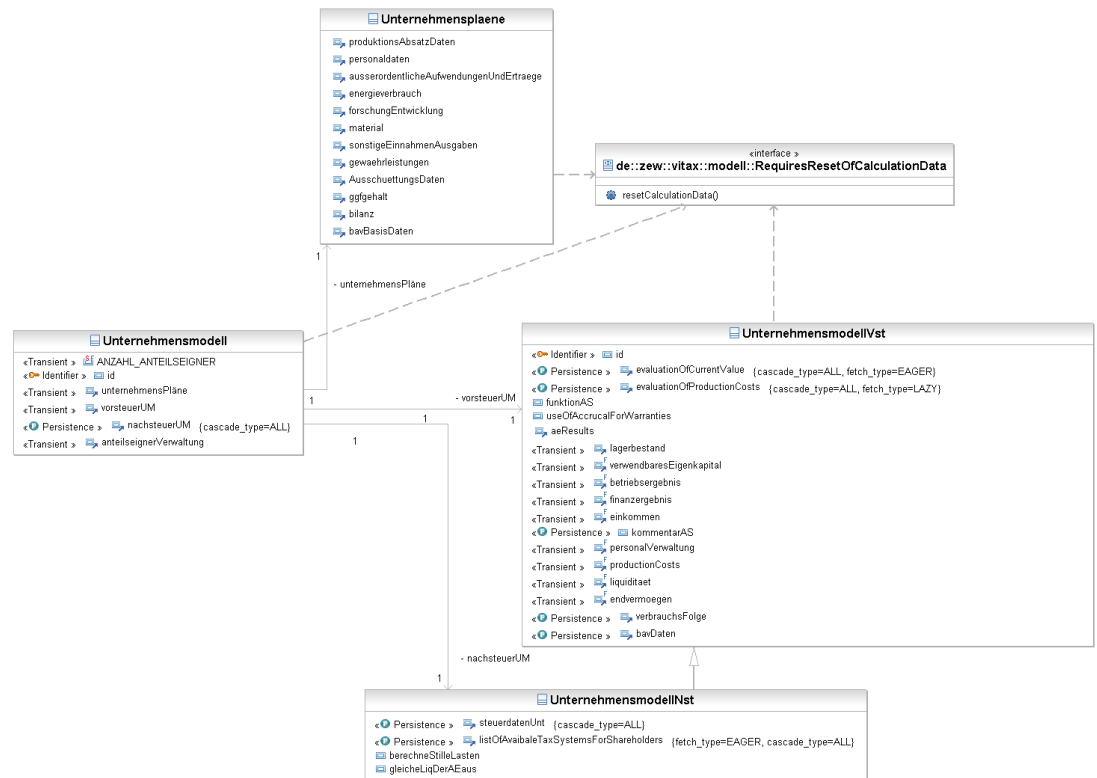
In der Klasse Unternehmensmodell (vgl. Abbildung 14) werden die ökonomischen Parameter (*Unternehmensplaene*) und Wahlrechte, die in beiden Fällen ausgeübt werden können (*UnternehmensmodellVst*) sowie steuerspezifische Optionen (*UnternehmesmodellNst*) vereint. Die Wahlrechte werden an das *UnternehmesmodellNst* vererbt, sodass nur dieses die steuerlich relevanten Daten enthält und in der Steuerdatenbank gespeichert werden muss. Der Rest entspricht dem rein ökonomischen Unternehmensmodell und muss separat gespeichert werden. Deswegen sind diese Attribute auch als *transient* markiert.

---

<sup>142</sup> Diese verweisen i. d. R. auf die Wirtschaftsdaten des Basismodells und werden bei Bedarf in ein sogenanntes Rumpfmodell via XML gespeichert.

Aufgrund dieser Struktur ist es nun recht einfach, z. B. verschiedene Branchen, Unternehmensgrößen oder andere Variationen für verschiedene Länder und Rechtsstände zu berechnen, da nun nur noch zwei Referenzen auszutauschen sind.

**Abbildung 14: Unternehmensmodell nach Zerlegung**



Somit muss im Idealfall nur der dritte Teil ohne die transienten Berechnungsergebnisse in der Datenbank hinterlegt werden. Als Problemfall stellte sich hier jedoch die Verknüpfung von Wirtschaftsgütern mit Abschreibungsregelungen heraus. Des Weiteren verweisen die Formeln für die Steuermodellierung häufig auf Bilanzobjekte oder Bestandteile der Gewinn-und-Verlust-Rechnung (GuV), die nicht originär in der Datenbank gespeichert werden. Daher müssen die zur Laufzeit des Programms eindeutigen Objektreferenzen in rekonstruier- und speicherbare Verweise umgewandelt werden.

**Abbildung 15: Übersicht über wichtige Aufzählungen zur Verknüpfung von ökonomischen und steuerlichen Daten**



Hierfür wurden verschiedene Aufzählungstypen (vgl. Abbildung 15) für die Bilanz, die GuV, für die Einkünfte und das Vermögen der Anteilseigner angelegt. Auf der linken Seite sind die möglichen Werte für die Aktiv- und Passivseite der Bilanz (z. B. *MACHINE\_1* für Maschine 1 oder deren entsprechenden Ersatz) zu sehen. In der Mitte sind die Substitute (z. B. *NET\_TURNOVER* für die Umsatzerlöse) für das Betriebs-, Finanz- und außerordentliche Ergebnis der Gewinn-und-Verlust-Rechnung zu sehen. Links stehen die möglichen Alternativen für die Vermögenswerte und Einkunftsarten der Anteilseigner. Diese bilden wiederum die Schlüsselwerte für assoziative Arrays (Maps) in denen, vereinfacht ausgedrückt, die Formeln, die Abschreibungsmethoden oder die Vorschriften zur Herstellungskostenermittlung als Wert hinterlegt sind (*evaluationOfCurrentValue* und *evaluationOfProductionCosts* - vgl. Abbildung 14). Abbildung 16 verdeutlicht nochmals die Auflösung dieser indirekten Verweise. Die entsprechenden Datenstrukturen für die Anteilseigner werden in der *anteilseignerVerwaltung* abgebildet.<sup>143</sup>

<sup>143</sup> Für detaillierte Informationen siehe Kapitel 5.3.7.2.4.



Abbildung 16: Auflösung der indirekten Verknüpfungen über das Unternehmensmodell



*BewertungAbschreibbaresBilanzObjekt* erweitert *BewertungBilanzObjekt* um eine Referenz auf die gewählte Abschreibungsmethode (*selectedDeprecation*). Instanzen beider Klassen werden in *evaluationOfCurrentValue* hinterlegt und dienen primär der Ermittlung der aktuellen Bewertung (Restbuchwerte) von Bilanzobjekten. Die Regeln, die für die Ermittlung der Herstellungskosten benötigt werden, sind in den *ValueProductionCosts* Objekten hinterlegt, die in *evaluationOfProductionCosts* verwaltet werden.

Als Beispiel für eine Klasse, die sowohl abschreibbar ist, als auch Bestandteil der Herstellungskosten sein kann, dient hier das *Wirtschaftsgut*. Ist ein Rückgriff auf den Restbuchwert notwendig, greift diese Klasse mittels ihrer eindeutigen Referenz (*key*)<sup>144</sup> z. B. *MACHINE\_1* auf den assoziative Array (*evaluationOfCurrentValue*) der entsprechenden Instanz<sup>145</sup> des *UnternehmensmodellVst* zu. Dadurch erhält es das zugehörige Bewertungsobjekt (*BewertungAbschreibbaresBilanzobjekt*). Ebenso erfolgt die Auflösung von *ValueProductionCosts* Objekten für die Herstellungskostenermittlung über *evaluationOfProductionCosts*.

<sup>144</sup> Der Schlüssel (*key*) kann eines der Aufzählungselemente aus Abbildung 15 sein. Im konkreten Fall muss dieser aus der Aufzählung *AktiveBilanzobjekte* stammen.

<sup>145</sup> Für den Fall ohne und mit Steuern gibt es separate Instanzen. *UnternehmensmodellNst* erweitert *UnternehmensmodellVst* (vgl. Abbildung 14), sodass im konkreten Fall keine Unterscheidung zwischen diesen Klassen notwendig ist.

#### 5.3.4 Herstellungskostenermittlung & Verbrauchsfolge

Die Regeln zur Bestimmung der Herstellungskosten und der Verbrauchsfolgeverfahren sind ein zentraler Aspekt der Gewinnermittlung, da nur beim Ansatz der Vollkosten keine direkten Auswirkungen auf das Ergebnis zu erwarten sind.<sup>146</sup> Die Berechnung wäre auch erfolgsneutral, wenn die Produktion in jeder Periode vollständig verkauft würde und somit kein Lagerbestand aufträte. In den Unternehmensplänen sind die exakten Daten des Produktions- und Absatzplanes hinterlegt. Zudem ist für die Bewertung des Lagerbestandes auch noch eine Verbrauchsfolge festzulegen. Diese stellt i. d. R., ebenso wie der Umfang der Bestandteile, ein steuerliches Wahlrecht dar.

Die im European Tax Analyzer erfassten Aufwendungen sind in Anteile für Produktion, Verwaltung und Vertrieb unterteilt und können mithilfe von Formeln den entsprechenden jeweiligen Regelungen angepasst und mit einbezogen werden (vgl. Abbildung 17). Alle nicht einbezogenen Posten oder Anteile werden in der Gewinn- und-Verlust-Rechnung direkt als Aufwand verbucht. Die Summe aus beiden Positionen ergibt den Gesamtaufwand pro Kostenträger. Die Herstellungskosten für nicht abgesetzte Produkte werden entsprechend der Verbrauchsfolge aktiviert. Hierbei wird jedoch unterstellt, dass die gesamte Produktion zunächst aufs Lager geht und danach die aktuelle Absatzmenge aus diesem entnommen wird. Die Gewinn- und-Verlust-Rechnung wird nach dem Umsatzkostenverfahren ermittelt.

Daher genügt es, den Produktions- und Absatzplan (Unternehmenspläne), die Vorschriften zur Ermittlung der Herstellungskosten (inkl. des Einbezugs in die Ermittlung des Aufteilungsschlüssels und dessen Formel) sowie das Verbrauchsfolgeverfahren zu speichern.

Bisher wurde hierfür der Lagerbestand inkl. dessen Bewertung gespeichert. Dieser bewertete Lagerinhalt ist aber nur ein temporäres Resultat und lässt sich aus den obigen Daten jederzeit wieder rekonstruieren. Somit wurde nun auf die Speicherung dieser Zwischenergebnisse verzichtet und nur noch die Art des Verfahrens zur Bewertung des Lagerbestandes (als Aufzählungstyp) gespeichert.

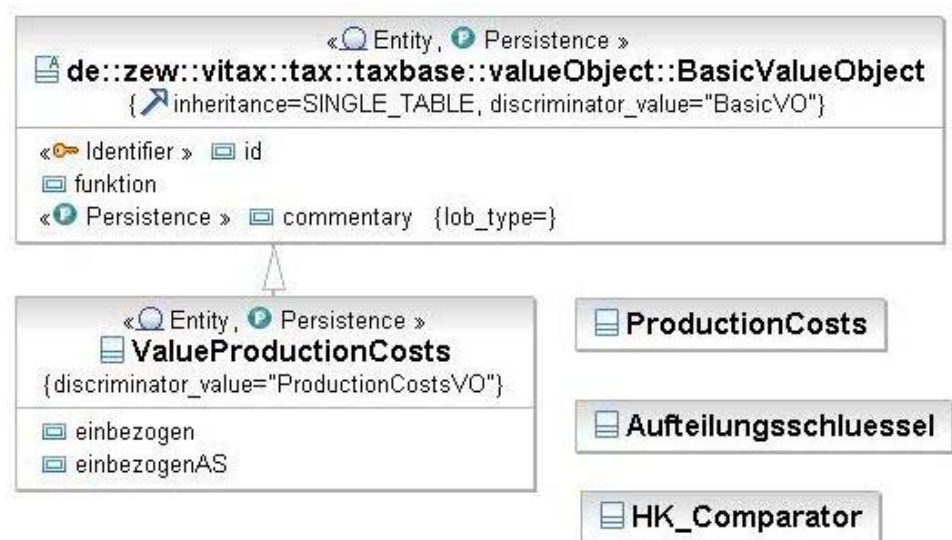
---

<sup>146</sup> Vgl. Bauch, G./Oestreicher, A., 1993, S. 131.

Abbildung 17: Dialog zur Eingabe der Regeln zur Ermittlung der Herstellungskosten

Bei der Speicherung der Vorschriften zur Ermittlung der Herstellungskosten stellt sich erneut (vgl. 5.3.3) das Problem der Referenzen hinsichtlich nicht in der Datenbank gespeicherter Objekte. Die obige Auflösung (vgl. Abbildung 15 und 16) wird daher wiederverwendet.

Abbildung 18: Bewertungsobjekt und Berechnungsmodule für die Herstellungskosten



*ValueProductionCosts* (vgl. Abbildung 18) erweitert das *BasicValueObject* um die Attribute *einbezogen* und *einbezogenAS*. Diese signalisieren, ob das referenzierte Ob-

jekt<sup>147</sup> in die Berechnung für die Herstellungskosten oder den Aufteilungsschlüssel einbezogen wird oder nicht. Eine Liste aller möglichen Bestandteile ist in Abbildung 19 zu sehen. Die Formel zur Ermittlung der Höhe des zu berücksichtigenden Aufwands wird in dem geerbten Attribut *funktion* gespeichert. Der Kommentar ermöglicht Erläuterungen zu dieser zu hinterlegen. Der Dialog zur Eingabe ist in Abbildung 17 dargestellt.

Eine Besonderheit stellt der Aufteilungsschlüssel (AS) dar. Aufgrund der Tatsache, dass der Einbezug der Bestandteile bereits in den Datenobjekten (*einbezogenAS - ValueProductionCosts*) hinterlegt ist (vgl. Abbildung 18) und sich somit leicht rekonstruieren lässt, genügt es, die Formel und den Kommentar des Aufteilungsschlüssels zu speichern. Daher wurden diese beiden Attribute aus der Klasse (*Aufteilungsschlüssel*) zur Berechnung des AS in das steuerlich relevante (Vor- und Nachsteuern)<sup>148</sup> Unternehmensmodell (vgl. Abbildung 14) verschoben. Ebenso dient *ProductionCosts* nur noch der Berechnung und der temporären Speicherung der Zwischenergebnisse.

Der *HK\_Comparator* (vgl. Abbildung 18) wird dazu benutzt, die Liste der Bestandteile alphabetisch zu sortieren, um somit eine rekonstruierbare Berechnungsreihenfolge zu gewährleisten.

---

<sup>147</sup> Im *UnternehmensmodellVst* ist ein assoziatives Array (*evaluationOfProductionCosts*) hinterlegt. Vgl. 5.3.3.

<sup>148</sup> *UnternehmensmodellNst* erbt dies von *UnternehmensmodellVst*; somit wird ein redundanter Quellcode vermieden.

Abbildung 19: Liste aller möglichen Bestandteile der Herstellungskosten und der benutzten Infrastruktur

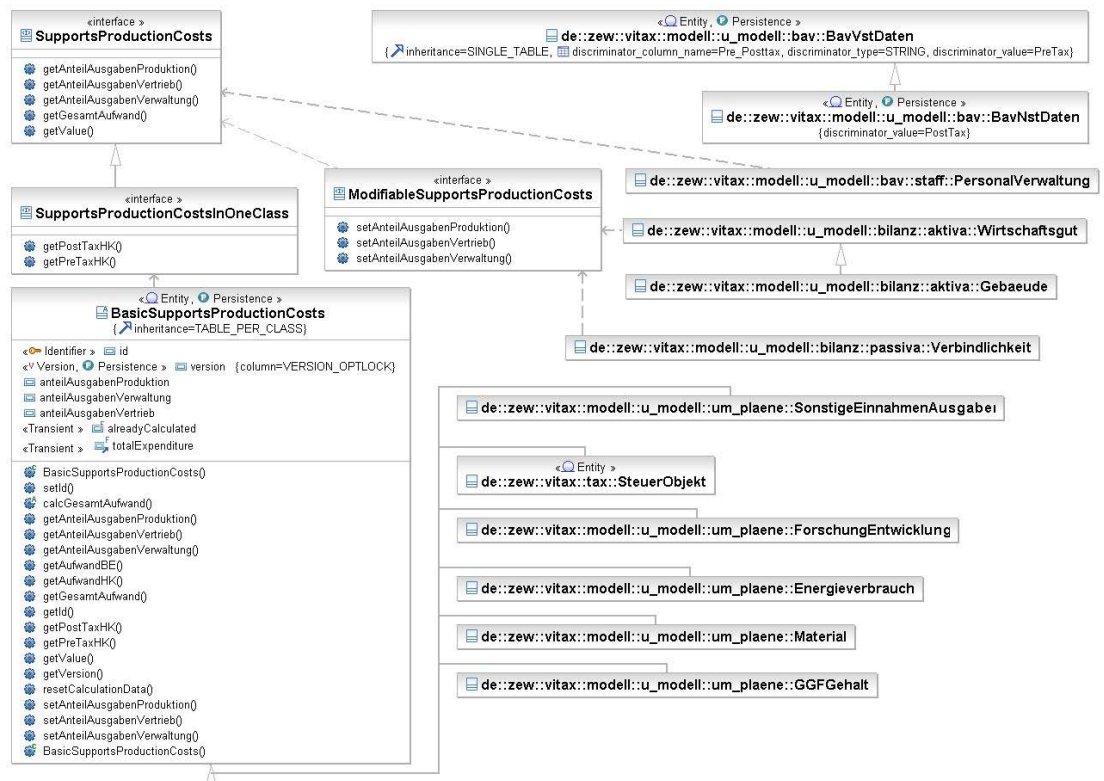


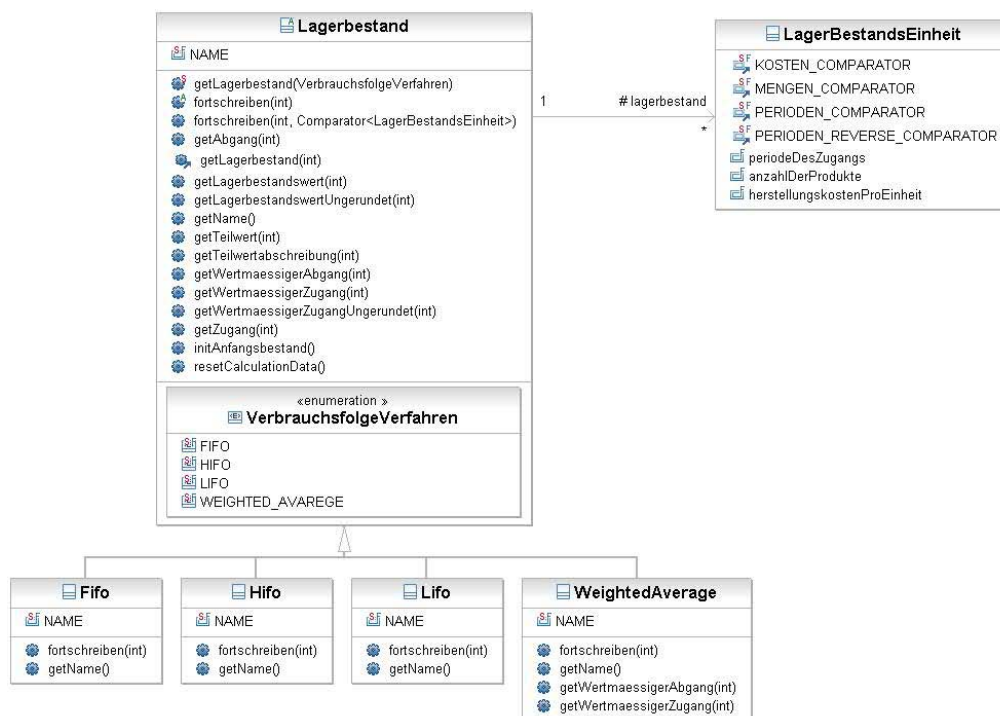
Abbildung 19 zeigt auf der linken Seite die Interface-Hierarchie, die zur Unterstützung der Herstellungskostenberechnung benötigt wird. *SupportsProductionsCosts* stellt hierbei das absolute Minimum dar. *SupportsProductionCostsInOneClass* ist eine bisher wenig benutzte Erweiterung, da sie erst spät eingeführt wurde. Sie soll für Bestandteile aus den Unternehmensplänen signalisieren, dass dieses Objekt sowohl die Bewertung für den Vorsteuer- als auch den Nachsteuerfall enthält. I. d. R. gibt *getValue()* dann den richtigen Wert für den aktuellen Berechnungsmodus zurück. *ModifiableSupportsProductionCosts* wird für die Eingabedialoge benutzt.

Diese Interfaces ersetzen die alte *HKSchnittstelle*.<sup>149</sup> Die Funktionen zur Berechnung der absoluten Anteile am Gesamtaufwand sind entfallen, da sich diese aus den jeweiligen Anteilen und dem Gesamtaufwand bereits berechnen lassen. Ebenso sind *calcProductionsCosts* und *getAufwandHK* entfallen, da sie bereits Bestandteil von *ValueProductionCosts* sind und somit via *getValue()* aufrufbar sind.

<sup>149</sup> Vgl. Stetter, T., VITAX, 2005, S. 124.

Die abstrakte Klasse *BasicSupportsProductionCosts* stellt eine rudimentäre Implementierung bereit. Auf der rechten Seite sind zurzeit alle möglichen Bestandteile abgebildet. Beispielsweise können auch Steuern wie die Grundsteuer mit einbezogen werden. Dies ist jedoch nur sinnvoll, wenn diese nicht selbst direkt oder indirekt von der Herstellungskostenermittlung abhängen, da ansonsten eine nicht berechenbare Wechselwirkung entsteht. Somit scheiden alle gewinnabhängigen Steuern aus.

**Abbildung 20: Klassendiagramm zur Berechnung des Lagerbestandes**



Zur Berechnung der Herstellungskosten wird die Liste aller möglichen Bestandteile durchlaufen. Wird ein Element einbezogen, wird anhand der hinterlegten Formel der zu berücksichtigende Anteil am Gesamtaufwand des Kostenträgers ermittelt und zur Summe der Herstellungskosten der aktuellen Periode hinzu addiert. Diese Summe wird anschließend durch die Anzahl der produzierten Güter dividiert und ergibt die Herstellungskosten pro Einheit. Diese werden dann zur Bewertung von Lagerzugängen verwendet.

Zur Ermittlung des Lagerbestandes wird für jede Periode eine Liste von aktuellen Zugängen bzw. Restbeständen aus vorherigen Perioden (*LagerBestandsEinheit*) verwaltet. Bei jedem Zugang zum Lager wird die Periode des Zugangs, die Menge und die Herstellungskosten pro Einheit in einer separaten Instanz vermerkt. Bei einem Abgang wird die Menge entsprechend reduziert, sodass sich aus den bisherigen Zugängen minus den bis dahin stattgefundenen Abgängen direkt der Restbestand der

aktuellen Periode ergibt. Bei einem Restbestand von 0 wird das Element in der folgenden Periode gelöscht. Ein evtl. vorhandener Lageranfangsbestand wird in Periode 0 als Zugang zum Lager verbucht. Reicht die noch vorhandene Menge nicht aus, so wird das nächste Element aus der Liste betrachtet und angemessen verringert. Der Endbestand (alle Elemente mit einer Menge größer null) der Vorperiode dient dann als Anfangsbestand der aktuellen Periode zuzüglich der aktuell produzierten Menge. Die neue Liste erstellt tiefe Kopien der Bestandteile, sodass die vorherigen Listen zum Zwecke der Datenausgabe unverändert bleiben.

Aufgrund dieser Vorgehensweise reicht es nun für die Umsetzung eines Verbrauchsfolgeverfahrens aus, die aktuelle Liste entsprechend zu sortieren. Deswegen übergeben die Spezialisierungen der abstrakten Klasse *Lagerbestand*, vereinfacht ausgedrückt, nur den zu verwendenden Sortierungsalgorithmus. Denn bei entsprechender Anordnung der Teile des Lagerbestandes ist garantiert, dass zuerst die tatsächlich infrage kommenden Elemente betrachtet werden. *Fifo* (First-in-first-out) benutzt z. B. den *PERIODEN\_COMPARATOR*, sodass die ältesten Bestände als Erstes betrachtet werden. Genau umgekehrt verhält es sich mit *Lifo* (Last-in-first-out) und dem *PERIODEN\_REVERSE\_COMPARATOR*. *Hifo* (Highest-in-first-out) nutzt den *KOSTEN\_COMPARATOR* und sorgt so dafür, dass die teuersten Bestandteile oben liegen. Der *MENGEN\_COMPARATOR* wird nicht benutzt, da hierfür zurzeit kein Verfahren existiert.

Nur das Durchschnittskostenverfahren benötigt eine spezielle Implementierung, da in diesem Fall zuerst für jede Periode der Mittelwert der Herstellungskosten ermittelt werden muss und die Reihenfolge der Zugänge keine Rolle spielt. Am Ende jeder Kalkulation für einen Zeitabschnitt enthält die Liste immer nur noch ein Element, mit der noch vorhandenen Menge und den neu ermittelten durchschnittlichen Herstellungskosten.

Als steuerlich relevanter Parameter muss nun nur noch eine Ausprägung (Element einer Aufzählung) des *VerbrauchsfolgeVerfahren(s)* (als Teil des Unternehmensmodells - vgl. Abbildung 14) gespeichert werden. Die konkrete Instanz einer der Subklassen wird mithilfe der statischen Methode *getLagerbestand* (*VerbrauchsfolgeVerfahren*) erzeugt.

Die Formeln zur Ermittlung des optional einbezogenen Aufwands sind in *ValueProductionCosts* Objekten hinterlegt, die vom steuerlichen Teil des Unternehmensmo-

dells (vgl. *evaluationOfProductionCosts* - Abbildung 16) gespeichert werden. Der Produktions- und Absatzplan sowie die funktionale Aufteilung, sind Teile des ökonomischen Modells und stellen somit keine steuerlichen Aspekte dar.

### 5.3.5 Abschreibungen

Um den Wertverlust von Wirtschaftsgütern durch Alterung oder Verschleiß zu erfassen, wird eine Absetzung für Abnutzung (AFA) für jede Periode ermittelt. Hierfür gibt es einige standardisierte Verfahren wie z. B. die lineare, degressive, digitale<sup>150</sup> oder auch die degressive Methode mit Übergang zur linearen.<sup>151</sup> Diese Wertminderungen können als Aufwand verbucht werden und reduzieren somit den zu versteuernden Gewinn. Die genauen Regelungen gibt der Gesetzgeber vor. Im Vergleichsmodell ohne Steuern wird i. d. R. linear über die wirtschaftliche Nutzungsdauer abgeschrieben.

Teilweise fließen diese Werte auch in die Ermittlung der Herstellungskosten und damit in die Bewertung des Lagerbestandes an Fertigerzeugnissen mit ein. Die Kosten für die Abnutzung einer Maschine werden z. B. i. d. R. anteilig den erzeugten Produkten als Teil der Anschaffungs- oder Herstellungskosten (AHK) zugerechnet. Dadurch wirkt dieser Aufwand erst beim Verkauf gewinnmindernd, da er vorher durch die Aktivierung in der Bilanz temporär neutralisiert wurde.

Somit können hier zwei, zum Teil auch gegenläufige, zeitliche Effekte auftreten. Ein hoher Anteil der Produktion, die den Lagerbestand erhöht, reduziert z. B. den Effekt von anfänglich erhöhten Abschreibungen.

Bisher lagen für jede Abschreibungsart separate Implementierungen und Datenmodelle vor. Aufgrund dieser verschiedenen Parameter müsste für jede Methode eine extra Tabelle im relationalen Schema erzeugt werden. Weil sich jeder Abschreibungsverlauf durch die Universalmethode abbilden lässt, wurden nun alle Methoden

---

<sup>150</sup> Aufgrund der geringen Verbreitung wurde dies nicht implementiert und ist auch wie alle anderen Spezialfälle durch die universelle Methode abbildbar. Vgl. Ihrig, H./Pflaumer, P., 2009, S. 150.

<sup>151</sup> Vgl. Ihrig, H./Pflaumer, P., 2009, S. 144 ff.



auf eine Spezialisierung der universellen Methode umgestellt, indem sie die reduzierten Eingabewerte<sup>152</sup> auf das komplexere Datenmodell übertragen.

*Abbildung 21: Eingabedialog der Universalmethode*

1	2	3	4	5	6	7	8	9
25.0	33.3333	50.0	100.0	0.0	0.0	0.0	0.0	0

Absolut: ☐ ☐ ☐ ☐ ☒ ☒ ☒ ☒ ☒

Relativ: ☒ ☒ ☒ ☒ ☐ ☐ ☐ ☐ ☐

AHK: ☒ ☐ ☐ ☐ ☒ ☒ ☒ ☒ ☒

RBW: ☒ ☒ ☒ ☒ ☐ ☐ ☐ ☐ ☐

Abschreibbar von den AHK sind (in %) : 100.0      Auflösung stille Reserven/Lasten in Jahren: 0

Kommentar

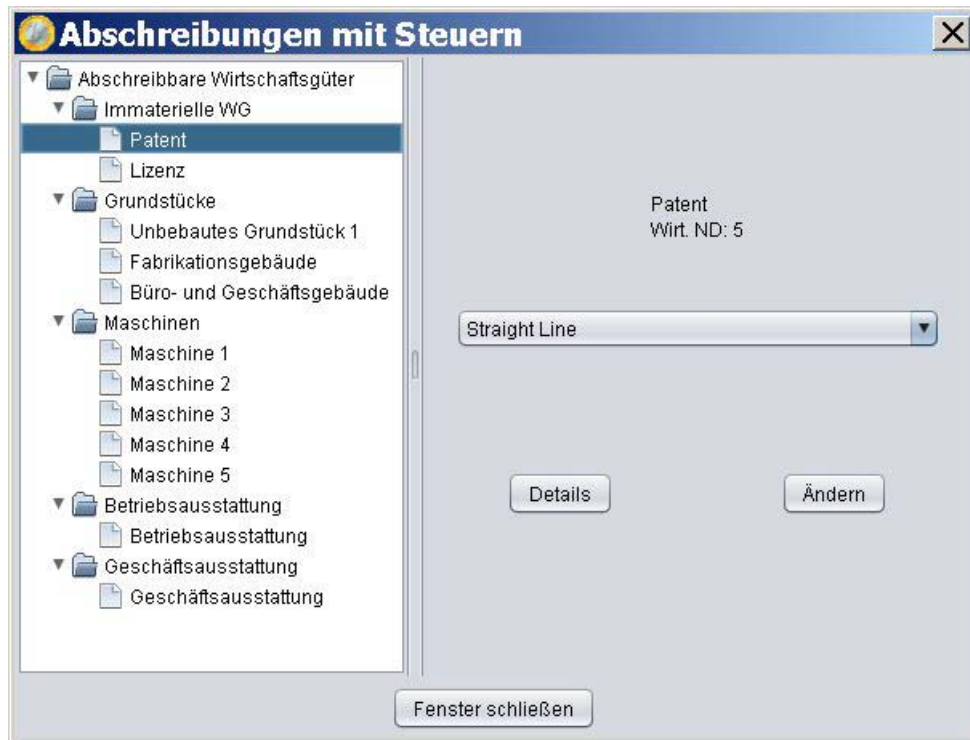
Speichern      Löschen

OK      Abbrechen

Die universale Methode definiert für jede Periode, welcher absolute oder relative Betrag in Abhängigkeit von den Anschaffungskosten oder dem Restbuchwert abzuschreiben ist (vgl. Abbildung 21). Daher ist es relativ einfach die benötigten Werte der anderen Abschreibungsarten zu berechnen und somit auf diese Methode abzubilden.

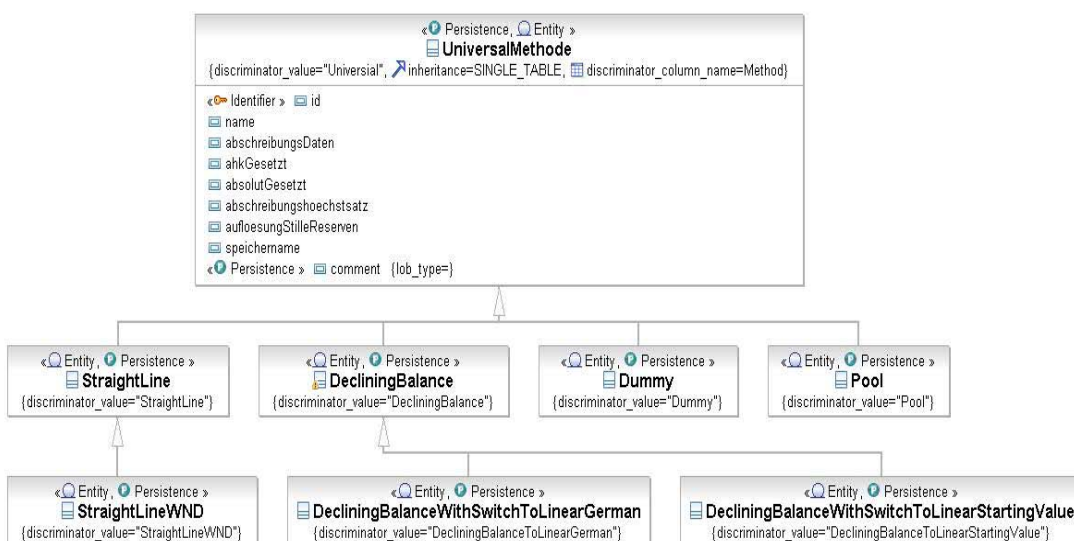
<sup>152</sup> Aus Vereinfachungsgründen wird der Benutzer z. B. bei der linearen Methode nur nach der Abschreibungsdauer gefragt, bzw. diese ist bereits durch die wirtschaftliche Nutzungsdauer festgelegt.

Abbildung 22: Auswahldialog der Abschreibungsmethoden



Dies hat den Vorteil, dass nur noch eine Abschreibungsmethode auf Korrektheit zu überprüfen ist und bei den anderen nur noch deren Abbildungsfunktion. Zudem können nun alle Varianten in einer Tabelle gespeichert werden. Zur Unterscheidung wird nur eine separate Spalte benötigt. Der Nachteil, dass nun bei sehr einfachen Abschreibungen mehr Informationen als unbedingt benötigt gespeichert werden, wird durch die Reduktion auf eine Tabelle und die stark reduzierte Fehlerwahrscheinlichkeit ausgeglichen. Deswegen wurde diese Lösung präferiert.

Abbildung 23: Vererbungshierarchie der verschiedenen Abschreibungsmethoden



Die Methode „lineare Abschreibung gemäß der wirtschaftlichen Nutzungsdauer“ ist als Spezialisierung der linearen Abschreibung implementiert. Dies dient primär der eindeutigen Unterscheidung, um zu dokumentieren, welche Methode der Gesetzgeber<sup>153</sup> vorgeschrieben hat. Es handelt sich genau genommen um einen Spezialfall. Daher muss nur die Dauer der Abschreibung angepasst werden. Deswegen wäre im Grunde keine eigene Klasse notwendig.

*Dummy* dient der Deaktivierung der Abschreibung und könnte genau genommen entfallen.<sup>154</sup> Dies ist bisher mit Rücksicht auf die Abwärtskompatibilität nicht erfolgt und ist auch programmatisch einfacher zu handhaben. Zudem ist so ein explizites Abschreibungsverbot auch besser erkennbar.

In der aktuellen Umsetzung unterscheiden sich *Pool* und *DecliningBalance* dadurch, dass die erste Methode immer bis zum Ende des Planungshorizontes einen Prozentsatz vom Restbuchwert abschreibt und somit nie Null<sup>155</sup> erreichen kann. Bei der zweiten ist die Dauer frei wählbar und am Ende wird der restliche Betrag ausgebucht. Aufgrund von mangelnder Codeduplizität wurde darauf verzichtet, die Poolmethode als Spezialisierung der degressiven Methode zu implementieren.

Die beiden degressiven Abschreibungen mit Übergang zur linearen Methode teilen sich mit der normalen degressiven Methode den gleichen Dialog und sind deshalb Teil der gleichen Vererbungshierarchie. Andernfalls könnte dies wie bei der Poolmethode entfallen.

Alle Spezialisierungen der Universalmethode besitzen keinerlei eigene Parameter. Diese werden bei der Eingabe direkt umgewandelt und für die Dialoge wieder zurück konvertiert. Somit enthält die Tabelle keine zusätzlichen, teilweise ungenutzten Parameter, sodass ein gravierender Nachteil der Strategie „Single Table“ entfällt.

Alle Abschreibungen erweitern nun die Universalmethode, sodass dort die Reste der früheren *DepreciationImp*<sup>156</sup> (vgl. Abbildung 4) aufgenommen wurden, die nun nicht mehr existiert. Ebenso wurde *Depreciation* entfernt und durch eine Verknüpfung zur

---

<sup>153</sup> Ansonsten wäre nicht klar, ob die Abschreibungsdauer nur durch Zufall mit der Nutzungsdauer übereinstimmt.

<sup>154</sup> Das Setzen des Verweises auf *null* würde bei entsprechender Behandlung ausreichen.

<sup>155</sup> Von rundungsbedingten Effekten einmal abgesehen.

<sup>156</sup> Vgl. Stetter, T., VITAX, 2005, S. 131.

Universalmethode ersetzt. Diese wurde durch das *BewertungsObjektAbschreibbares-BilanzObjekt* indirekt realisiert. Es enthält neben der gewählten Abschreibungs-  
methode auch die temporären Restbuchwerte und ist eine Spezialisierung von *BewertungsObjektBilanzObjekt* (vgl. Abbildung 16).

### 5.3.6 Betriebliche Altersversorgung

Als zusätzlichen Gehaltsbestandteil ist im European Tax Analyzer eine betriebliche Altersversorgung vorgesehen. Jeder Mitarbeiter erwirbt i. d. R. jedes Jahr einen gewissen Anspruch auf Zahlung einer Rente. Je nach Durchführungsform führt dieser Anspruch aber erst bei Eintritt in den Ruhestand zu liquiditätswirksamen Auszahlungen. Somit sind die Liquiditätswirkungen der betriebliche Altersversorgung im Gegensatz zu den Löhnen zeitlich versetzt; Aufwendungen fallen hingegen zeitgleich an. Dieser temporale Effekt ist im Hinblick auf Steuerbelastungsvergleiche besonders interessant. Ebenso wie die Löhne können diese Aufwendungen Bestandteil der Herstellungskosten sein. Durch die Aktivierung in der Bilanz kann es passieren, dass sich die zeitlichen Auswirkungen zum Teil wieder neutralisieren. Zudem handelt es sich durch die langen Anspardauern i. d. R. um recht hohe Beträge. Diese können somit einen deutlichen Einfluss auf das Ergebnis haben. Insbesondere beim Vergleich zwischen dem ökonomischen Unternehmensmodell und dem steuerlichen Modell kann es zu großen Unterschieden kommen, da steuerliche Rückstellungen häufig nachteilige Berechnungsannahmen unterstellen.<sup>157</sup> Dieser Bewertungsunterschied wird durch den Ansatz einer stillen Last (oder Reserve) beim Endvermögensvergleich wieder ausgeglichen, sodass sich ausschließlich die temporalen Effekte in der Gesamtbetrachtung auswirken.

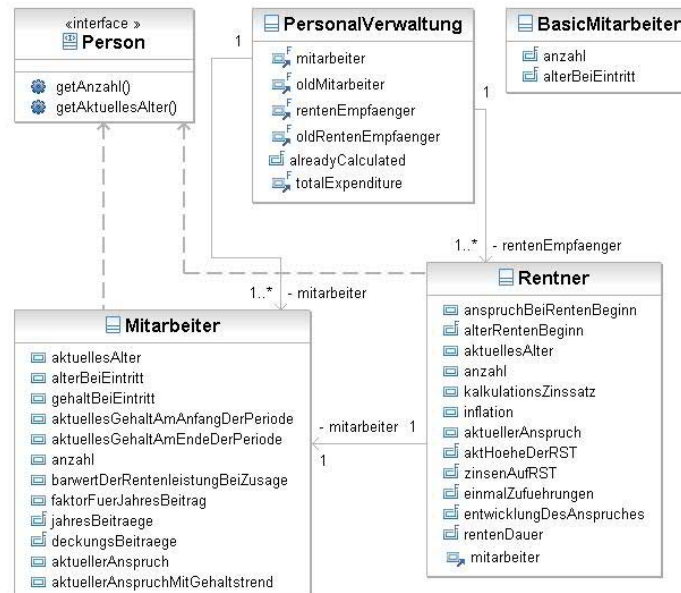
Für die persistente Speicherung im Unternehmensmodell wurde die Klasse *BasicMitarbeiter* (vgl. Abbildung 24) vorgesehen. Sie speichert nur die Anzahl der Mitarbeiter pro Altersstufe. Zur Verwaltung der temporären Zwischenergebnisse wurden die Klassen *Mitarbeiter* und *Rentner*, die beide das *Interface Person* implementieren, realisiert. Die Aktualisierung (z. B. das Fortschreiben des Alters) der Instanzen und

---

<sup>157</sup> In Deutschland ist zum Beispiel ein Abdiskontierungsfaktor in Höhe von 6 % (vgl. Vgl. § 6a Absatz 3 Satz 3 EStG) sowie eine Altersgrenze von 28 Jahren (vgl. § 6a Absatz 2 Nr. 1 EStG) festgeschrieben.

deren Administration erfolgt über die *PersonalVerwaltung*. Diese erzeugt auch zu Beginn jeder Berechnung aus der Liste der *BasicMitarbeiter* die Liste der Mitarbeiter. Die Liste der *Rentner* ist, aufgrund von Annahmen, zu Beginn leer.

Abbildung 24: Personalverwaltung des Variable International Tax Analyzer



Die Einstellungen, die sich für beide Fälle (mit und ohne Steuern) nicht unterscheiden dürfen, werden in der Klasse *BavBasisDaten* (vgl. Abbildung 25) gespeichert. Hierunter fallen z. B. die Art der Zusage und grundlegende Rahmendaten wie z. B. das Rentenalter oder die Altersstufe für Neueinstellungen. *Durchführungsform* und *Berechnungsmethode* werden durch entsprechende Aufzählungen abgebildet. Die konkreten Ausprägungen werden von *BavVstDaten* verwaltet. *BavNstDaten* erweitert diese um Parameter für steuerliche Sachverhalte<sup>158</sup> und enthält somit alle für die Datenbank relevanten Informationen. Beide enthalten auch eine temporäre Referenz auf das konkrete Berechnungsmodul und die Regeln zum Einbezug in die Herstellungskosten (*valueProductionCosts*).

<sup>158</sup> Dies sind z. B. der zu verwendete Zinssatz, die Altersgrenze und die Frage, ob Gehaltssteigerungen (Trend) oder die Inflation bei der Berechnung berücksichtigt werden dürfen.

Abbildung 25: Datenmodell der betrieblichen Altersvorsorge



Die abstrakte Klasse *BavAnsparArten* (vgl. Abbildung 26) enthält bereits die notwendigen Daten und führt die eigentliche Berechnung durch. Hierbei wird z. B. die Methode *brechneAnspruchProJahr* aufgerufen. Die konkrete Instanz einer der Unterklassen (*BAV\_GehaltsabhaengigeZusage*, *BAV\_Festzusage*, *BAV\_Beitragszusage*) gibt dann den für diese Form der Zusage spezifischen Wert<sup>159</sup> zurück. Somit müssen für die Spezialisierungen nur noch wenige Methoden überschrieben und validiert werden.

Die Berechnung ermittelt auch die Höhe der notwendigen Rückstellungen, der liquiditätswirksamen Auszahlungen<sup>160</sup> und die Höhe des Gesamtaufwands. Die Aufteilung zwischen Herstellungskosten und dem Betriebsergebnis richtet sich nach der funktionalen Aufteilung der Löhne und Gehälter. Alle diese Daten sind temporäre Resultate und sind somit nicht zu speichern.

<sup>159</sup> Bei *brechneAnspruchProJahr* ist dies z. B. die Höhe des Anspruchs, den sich ein Mitarbeiter in einem Jahr erwirbt. Bei der gehaltsabhängigen Zusage ist dies i. d. R. ein Prozentsatz des aktuellen Gehalts. Bei der Beitragszusage der Betrag, der für dieses Jahr zugesagt worden ist. Bei der Festzusage ist dieser Betrag unabhängig von der Periode.

<sup>160</sup> Dies können Rentenzahlungen oder Zuführungen zu einer Pensionskasse sein.

Abbildung 26: Polymorphismus der Berechnungsmodule

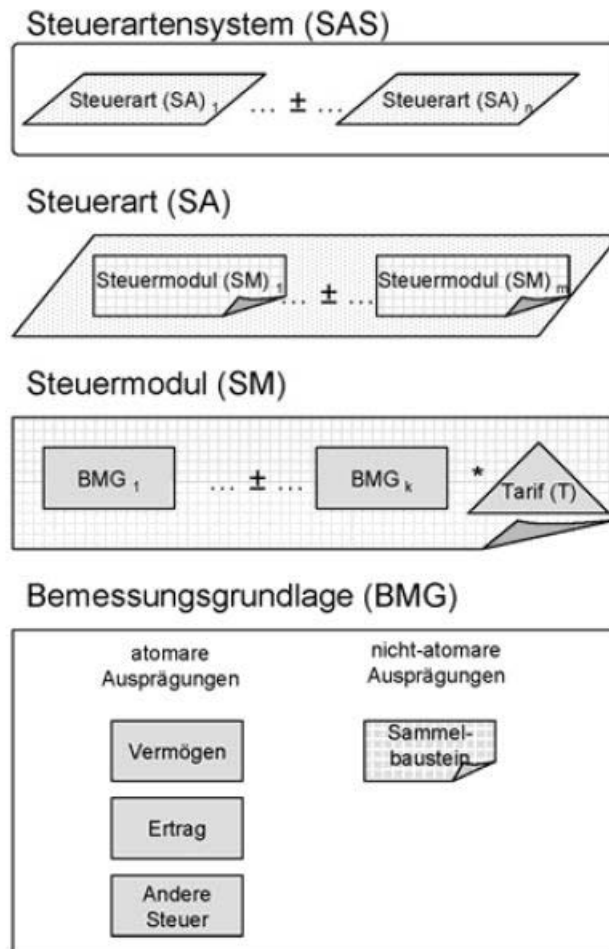


### 5.3.7 Steuern

#### 5.3.7.1 Modellierung des Steuerartensystems

Ein Steuersystem setzt sich zumeist aus mehreren Steuerarten zusammen. Steuerarten bestehen grundsätzlich aus einer Bemessungsgrundlage und einem Tarif. Die Bemessungsgrundlage (BMG) wird dadurch abgebildet, dass die verschiedenen Bestandteile mit mathematischen Operatoren verknüpft werden. Für das Datenbankschema ist es sinnvoll, das bisherige Metamodell (vgl. Abbildung 27) auf mögliche Anpassungen zu überprüfen, da dieses den elementaren Bestandteil darstellt.

Abbildung 27: Metamodell zur Modellierung des Steuerartensystems



Quelle: Stetter, T., VITAX, 2005, S. 61.

Die Modulverwaltungen von *Steuerart* und *Steuermodul* unterscheiden sich primär durch den Tarif, da als Module nur *Sammelbausteine* erlaubt sind. Ebenso wird die Modulverwaltung zurzeit über den *Sammelbaustein* realisiert. Dies ermöglicht auch auf der Ebene der Module eine rekursive Schachtelung, sodass Ebene 2 aus Abbildung 27 entfallen kann.<sup>161</sup> Daher ist es sinnvoll, diese zu vereinheitlichen.

Der Vorteil der gewählten Lösung liegt neben der Einsparung eines Dialoges auch darin, dass auf diese Weise nur noch eine Verwaltung, anstatt zweier, zu validieren und in der Datenbank abzubilden ist. Durch die mögliche rekursive Schachtelung

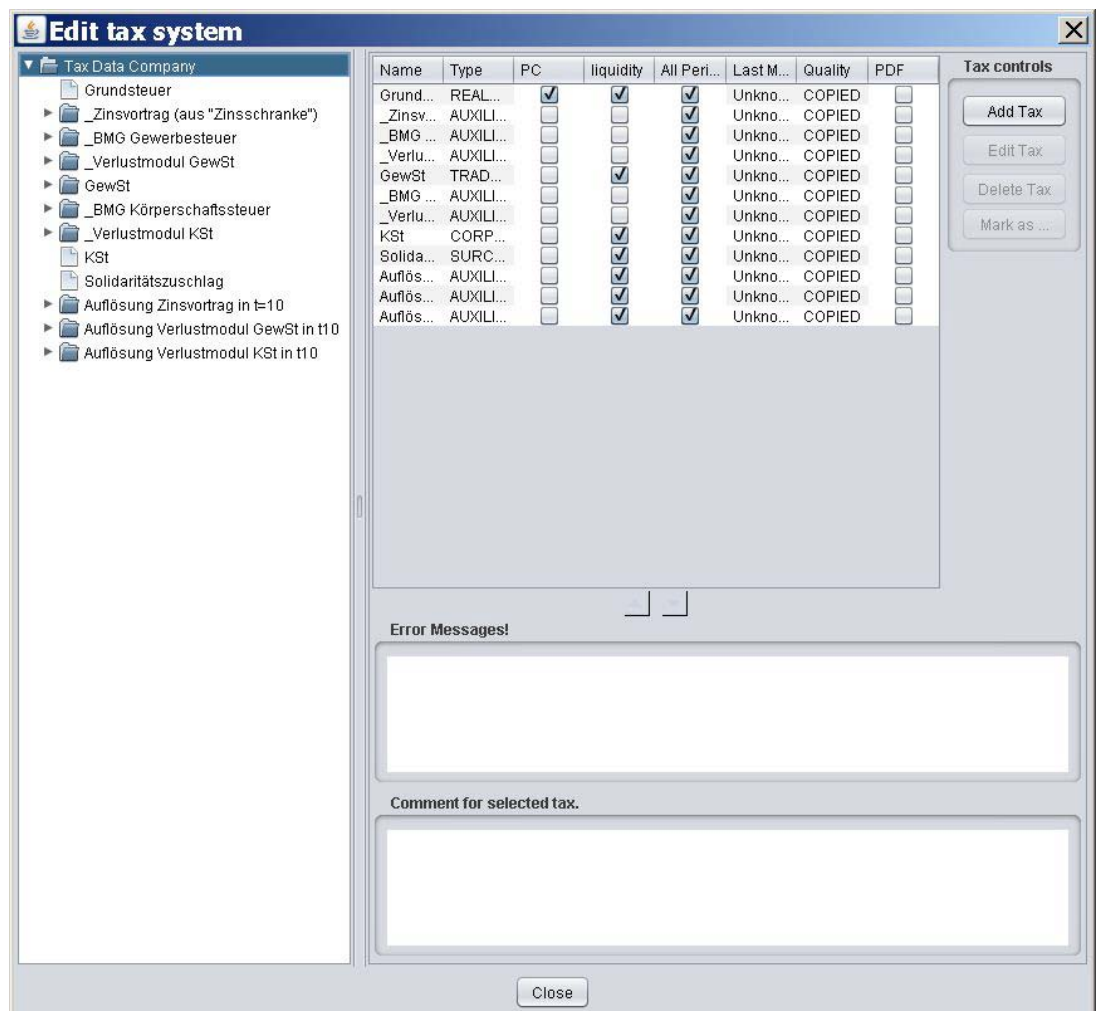
<sup>161</sup> Wobei auf Ebene 3 Steuermodul durch Steuerart zu ersetzen wäre.



und die gleichzeitige Verwendung eines linearen Tarifs mit 100 %, <sup>162</sup> kann die vorherige Modellierung bei Bedarf nachgebildet werden. <sup>163</sup>

Für die Zwecke der vereinfachten Recherche durch das Web-Interface wurde eine Klassifizierung (vgl. Abbildung 30 *TypeOfTax*) eingeführt. Hiermit soll jeder Steuer eine bestimmte Steuerart zugewiesen werden. Dies ist alleine schon dadurch hilfreich, da es nun möglich ist, Hilfsrechnungen eindeutig zu markieren. Der eigentliche Zweck besteht darin, die Recherche von verschiedenen Typen zu ermöglichen und zu vereinfachen. Allerdings kann dies auch für eine automatische Steuerartenrechnung benutzt werden.

**Abbildung 28: Dialog zum Bearbeiten von Steuern**



<sup>162</sup> Durch einen Tarif von 100 % wird das Ergebnis unverfälscht weitergereicht.

<sup>163</sup> Dies ist m. E. nur dann sinnvoll, wenn aus Gründen der Übersichtlichkeit eine zusätzliche Ebene gewünscht ist.

Des Weiteren wurden auch eine Versionierung<sup>164</sup> und eine Historie (vgl. 3.2.3.2) vorgesehen. In Abbildung 28 ist auch ersichtlich, dass die Möglichkeit besteht zu spezifizieren, in welchen Perioden eine Steuer berechnet wird, ob sie in die Herstellungskosten (Production Costs (PC)) einbezogen werden kann oder ob sie sich auf die Liquidität des Unternehmens auswirkt. Das Klassendiagramm ist in Abbildung 30 ersichtlich und verdeutlicht nochmals die zu speichernden Details.

#### 5.3.7.2 Bemessungsgrundlagen

Die Steuer verwaltet ihre Module durch einen Sammelbaustein. Zurzeit gibt es sieben verschiedene Bemessungsgrundlagen (Bilanz, Gewinn-und-Verlust-Rechnung<sup>165</sup>, Vermögen und Einkünfte des Anteilseigners, Rückgriff auf andere Steuern, Verlustverrechnung und der universelle Sammelbaustein). Diese kann man grob in zwei Gruppen unterteilen. Die erste greift i. d. R. direkt auf Daten des Unternehmensmodells inkl. der Anteilseigner zurück und ist formelbasiert, während die zweite primär der Steuerberechnung dient.

Als einzige Bemessungsgrundlage (BMG) enthält der Sammelbaustein auch einen Tarif (vgl. 5.3.7). Der Tarif dient aber nicht nur zur Abbildung von Steuertarifen, sondern stellt eine universelle Möglichkeit zur Modifikation der vorherigen Resultate dar. Die Exklusivität ist ausreichend, da jede BMG durch alleinige Einbettung in einen Sammelbaustein um einen Tarif erweitert werden kann.

##### 5.3.7.2.1 Sammelbaustein und Tarif

Dieses Element ist die zentrale Komponente der Steuermodellierung. Es speichert eine Liste von Bemessungsgrundlagen und deren mathematische Operatoren. Der Sammelbaustein kann auch weitere Ausprägungen dieses Bausteins enthalten, wodurch eine tiefe Schachtelung möglich ist. Diese Komponente stellt sozusagen Kno-

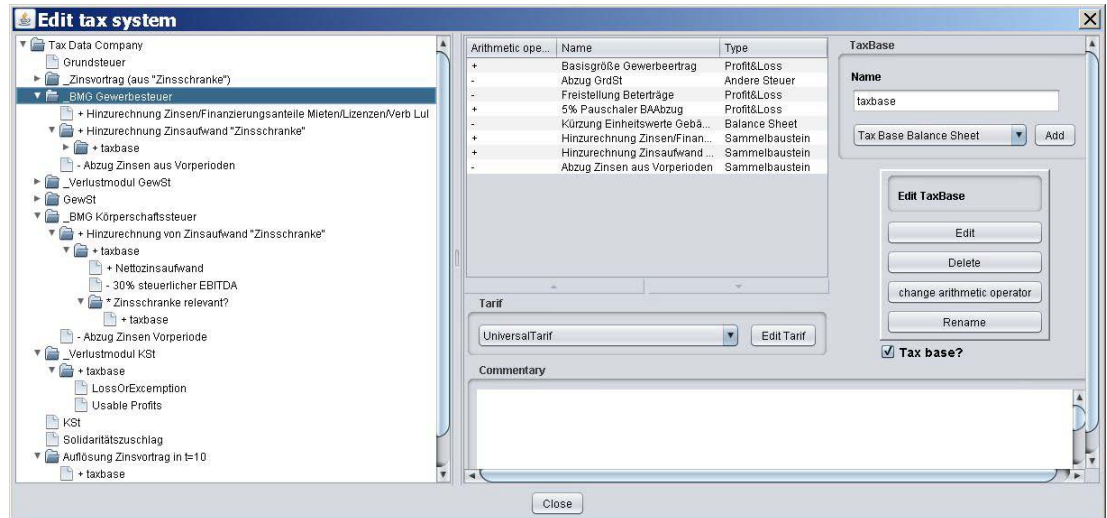
---

<sup>164</sup> Wird zurzeit nicht explizit genutzt. Dies könnte sinnvoll sein, falls sich die Kollisionen durch gleichzeitiges Bearbeiten eines Landes in der Zukunft häufen. Somit könnte die Granularität der Sperren erhöht werden und kollisionsbedingter Datenverlust und die Wahrscheinlichkeit eines konkurrierenden Zugriffs deutlich verringert werden (vgl. 4.2).

<sup>165</sup> In der Bilanz wird nur das Resultat der Gewinn-und-Verlust-Rechnung in Form des Jahresüberschusses ausgewiesen. Werden Zwischenresultate benötigt, ist daher eine eigene Bemessungsgrundlage notwendig.

ten bzw. Verzweigungen eines Baumes dar, während die anderen Bemessungsgrundlagen<sup>166</sup> eher Blättern entsprechen.

Abbildung 29: Benutzeroberfläche des Sammelbausteins (inkl. Steuerbaum links)



Der Verzicht auf einen Tarif war auch der einzige Unterschied zwischen der bisherigen Modulverwaltung des Steuerobjektes und dieser Komponente. Somit wurde, aus Vereinfachungsgründen und wegen der Einsparung eines separaten Dialogs, die Liste<sup>167</sup> des Steuerobjektes (vgl. Abbildung 8) durch eine 1:1-Beziehung zum Sammelbaustein (vgl. Abbildung 30) ersetzt.

Zusätzlich wurde die lose Bindung zwischen Operationen und Bemessungsgrundlage durch eine Hilfsklasse *BMG\_Baustein* ersetzt. Diese war zuvor durch zwei verschiedene Listen realisiert (vgl. Abbildung 8). Hierbei wurde erwartet, dass zugehörige Komponenten den gleichen Index besaßen. Dieser Baustein wäre eigentlich für das Immutable-Pattern<sup>168</sup> prädestiniert gewesen. Dies konnte jedoch aufgrund von Restriktionen eines Persistenz-Frameworks<sup>169</sup> nicht umgesetzt werden.

Abbildung 30 verdeutlicht nochmals die neue Struktur. Die Option *taxbase* signalisiert, dass die aktuelle Komponente der Bemessungsgrundlage einer Steuer ent-

<sup>166</sup> Die Verlustverrechnung enthält Referenzen auf zwei Sammelbausteine, sodass sie kein Blatt im eigentlichen Sinne darstellt, sondern eher eine Astgabel.

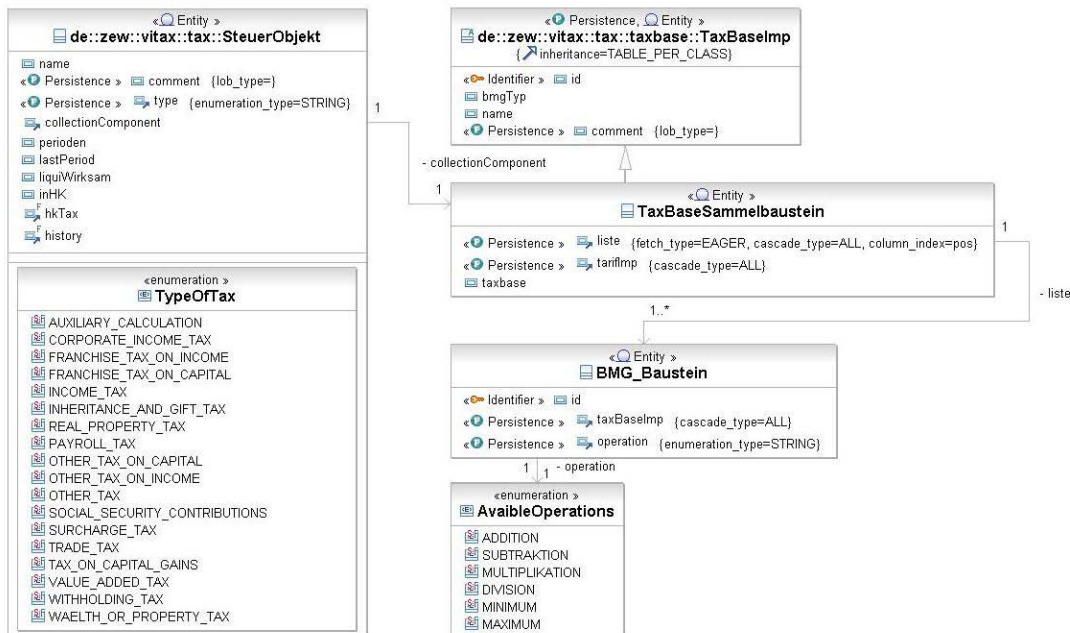
<sup>167</sup> Vgl. Stetter, T., VITAX, 2005, S. 147.

<sup>168</sup> Nach der Erzeugung einer Instanz einer Klasse können die zugehörigen Attributwerte nicht mehr verändert werden. Wird ein Objekt mit anderen Werten benötigt, muss ein neues erzeugt werden. Dies hat unter anderem den Vorteil, dass beim Erstellen von Kopien nur die Referenz kopiert werden muss, da sich das Zielobjekt nicht ändern kann.

<sup>169</sup> Hibernate unterstützt keine finalen Attribute. Auf die Implementierung von Änderungsmethoden kann aber verzichtet werden, da die direkte Manipulation der Attribute durch Hibernate möglich ist.

spricht. Dies ist hilfreich, um in Zukunft die isolierte und automatisierte Evaluierung von Bemessungsgrundlagen zu ermöglichen.

**Abbildung 30: Zusammenspiel zwischen Steuerobjekt, TaxBaseSammelbaustein und BMG\_Baustein**



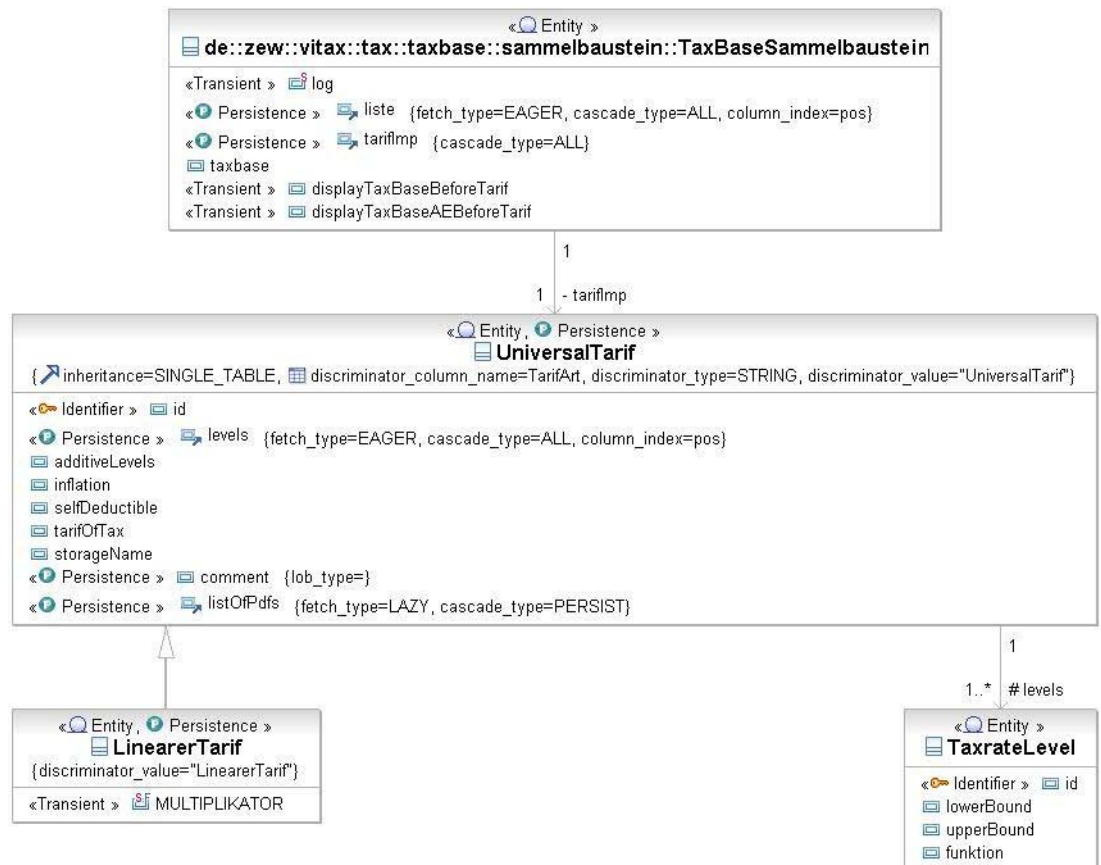
Des Weiteren speichert der Sammelbaustein den zu verwendenden Tarif. Dieser muss aber nicht unbedingt ein Steuertarif sein, sondern kann auch für andere Berechnungen benutzt werden. In vielen Fällen ist jedoch gar keine Veränderung gewünscht, sodass in diesen Fällen häufig ein linearer Tarif, mit 100 %, zum Einsatz kommt.

Im einfachsten Fall wird das Ergebnis der Bemessungsgrundlage, z. B. der aktuelle Gewinn, mit dem Steuersatz (i. d. R. ein prozentualer Wert) multipliziert. Dies entspricht dann einem linearen Tarifverlauf. Es sind jedoch weitaus kompliziertere Gestaltungen möglich, sodass eine flexible, universelle Lösung benötigt wird.

Bisher wurde noch zwischen einem linearen, progressiven (i. d. R. stufenweise ansteigender Steuersatz) und einem universellen Tarif (vgl. Abbildung 9) unterschieden. Diese waren bisher entsprechend separat implementiert. Sie stellen jedoch alle Untermengen des universellen Tarifs dar. Insbesondere der progressive Tarif war auch von den Eingabemasken her fast identisch, sodass er zugunsten einer klareren Struktur gänzlich entfallen ist und durch den universellen ersetzt wurde. Ebenso wurde der lineare Tarif als Spezialisierung des Universaltarifs implementiert, sodass nur die vereinfachte Eingabemaske übrig geblieben ist und die Umrechnung der Eingabepa-

parameter hinzugefügt wurde. Dies hat den Vorteil, dass analog zu den Abschreibungsmethoden nun nur noch eine Implementierung eines Tarifes zu validieren ist. Ebenso wird für die Abbildung in die relationale Datenbank nur noch eine Tabelle benötigt. Daher wurde als Vererbungsstrategie auch Single-Table (vgl. Abbildung 31) gewählt.

Abbildung 31: Implementierung des Tarifs



Wie in Abbildung 32 ersichtlich, besteht der universelle Tarif aus einer oder mehreren Stufen (*TaxrateLevel* - vgl. Abbildung 31), für die einzelne Formeln hinterlegt werden können. Die Variable  $x$  enthält das Ergebnis der im Sammelbaustein hinterlegten Bemessungsgrundlagen. Des Weiteren kann mit der Variablen  $xr$  auf den Wechselkurs des jeweiligen Landes zugegriffen werden. Dies ist notwendig um

Fremdwährungsbeträge<sup>170</sup> umzurechnen. Die sichtbaren Grenzen entsprechen dem Wertebereich des Typen *long*<sup>171</sup> in Java. Beim Hinzufügen einer Stufe muss diese Spanne entsprechend verteilt werden. Die untere Grenze der ersten Stufe ist frei wählbar, während die obere Barriere immer von der unteren der nächsten Stufe bestimmt wird. Das obere Limit der letzten Ebene entspricht immer dem Wert *Long.MAX\_VALUE*. Sollte dies nicht gewünscht sein, kann ein separates Niveau mit der Formel 0 eingefügt werden. Der lineare Tarif enthält nur eine Stufe mit den negativen und positiven Extremwerten als Grenze. Als Formel wird der eingetragene Prozentsatz<sup>172</sup> hinterlegt.

Abbildung 32: Eingabemaske des Universaltarifs

From (inclusive)	To (exclusive, if next level)	Formula
-9.223.372.036.854.000	9.223.372.036.854.000	x

Buttons: Add Level, Remove last Level

Manage PDF's: Open, Add, Delete

Options:
   
☐ Stufentarif (additive Stufen)
   
☐ Inflationierung
   
☐ in sich abzugsfähige Steuer
   
☐ Tatsächlicher Tarif der Steuer?

Commentary: [Text area]

Buttons: Save, Delete, OK

<sup>170</sup> Der Universaltarif rechnet das Ergebnis der Liste der Bemessungsgrundlagen in die jeweilige ländertypische Währung um, sodass direkt die kodifizierten Stufengrenzen und Freibeträge in der jeweiligen Landeswährung eingegeben werden können. Das Resultat wird dann wieder in die Modellwährung (Euro) zurückgerechnet. In seltenen Fällen kann es aber sein, dass die Bemessungsgrundlage bereits in Fremdwährung vorliegt. Um die dann unnötiger Weise vorgenommene Umrechnung rückgängig zu machen, kann der Wechselkurs Teil der Formel sein.

<sup>171</sup> Das Ergebnis wird mit *double* ermittelt, das einen deutlich größeren Wertebereich als *long* hat, sodass Overflows nahezu ausgeschlossen sind. Zudem ist es recht unwahrscheinlich, dass der Wertebereich von ca. 9 Trillionen ( $2^{63} - 1$ ) jemals ausgeschöpft wird.

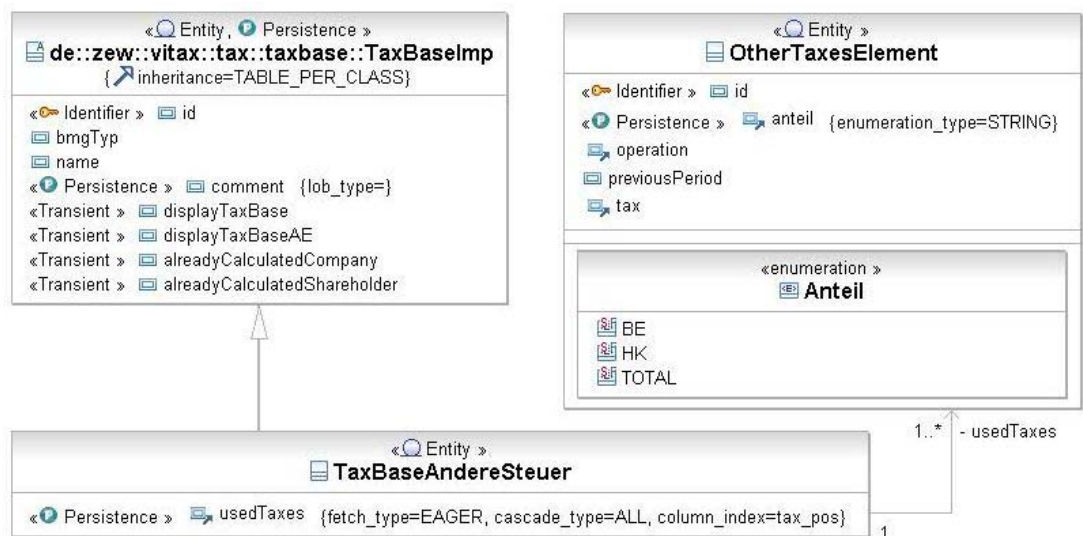
<sup>172</sup> Bei 25 % lautet die Formel z.B.:  $x * 0,25$ .

Des Weiteren gibt es noch Optionen, welche die genaue Anwendung der einzelnen Stufen und Teiltarife bestimmen. Der Stufentarif addiert die Resultate einzelner Stufen, wobei die Bemessungsgrundlage auf jedes Level entsprechend verteilt wird. Ohne die Wahl dieser Alternative würde der gesamte Betrag auf das Intervall entfallen, in dessen Wertebereich er liegt. Ansonsten gibt es noch die Möglichkeit die Grenzen zu inflationieren und In-Sich-Abzugsfähigkeit auszuwählen. Zusätzlich besteht noch die Möglichkeit, diesen Tarif als Tarif einer Steuer zu markieren. Dies dient primär Recherchezwecken für das Webinterface und ermöglicht es Hilfsrechnungen auszubilden, da nicht immer der oberste Tarif auch dem Steuertarif entspricht.

#### 5.3.7.2.2 Andere Steuern

Häufig besteht die Notwendigkeit, bei der Berechnung einer Steuer auf die Ergebnisse anderer Steuern zurückzugreifen. Bei der Einkommensteuer in Deutschland muss z. B. das Ergebnis der Kirchensteuer mit in die Kalkulation einbezogen werden, da diese als Sonderausgabe<sup>173</sup> abziehbar ist. Dieser Rückgriff kann horizontal, d. h. auf bereits vorher berechnete Steuern der aktuellen Periode, oder vertikal, d. h. auf vorhergehende Perioden, erfolgen. Zudem besteht so die Möglichkeit auch Zwischenrechnungen einzubinden, die i. d. R. als nicht liquiditätswirksame Steuern modelliert werden.

Abbildung 33: Klassendiagramm andere Steuern



<sup>173</sup> § 10b Absatz 1 EStG.



In *OtherTaxesElement* (vgl. Abbildung 33) wird gespeichert, für welche Periode eine Steuer ausgewertet wird, welcher Anteil betrachtet wird und welche mathematische Rechenoperation auf das Ergebnis angewendet werden soll. Der Anteil ist primär dann interessant, wenn die referenzierte Steuer in die Ermittlung der Herstellungskosten einfließt. *TaxBaseAndereSteuern* enthält eine Liste (*usedTaxes*) dieser Elemente und erweitert die abstrakte Basisklasse *TaxBaseImp*. Der entsprechende Eingabedialog ist in Abbildung 34 zu sehen.

**Abbildung 34: Benutzeroberfläche zum Rückgriff auf andere Steuern**



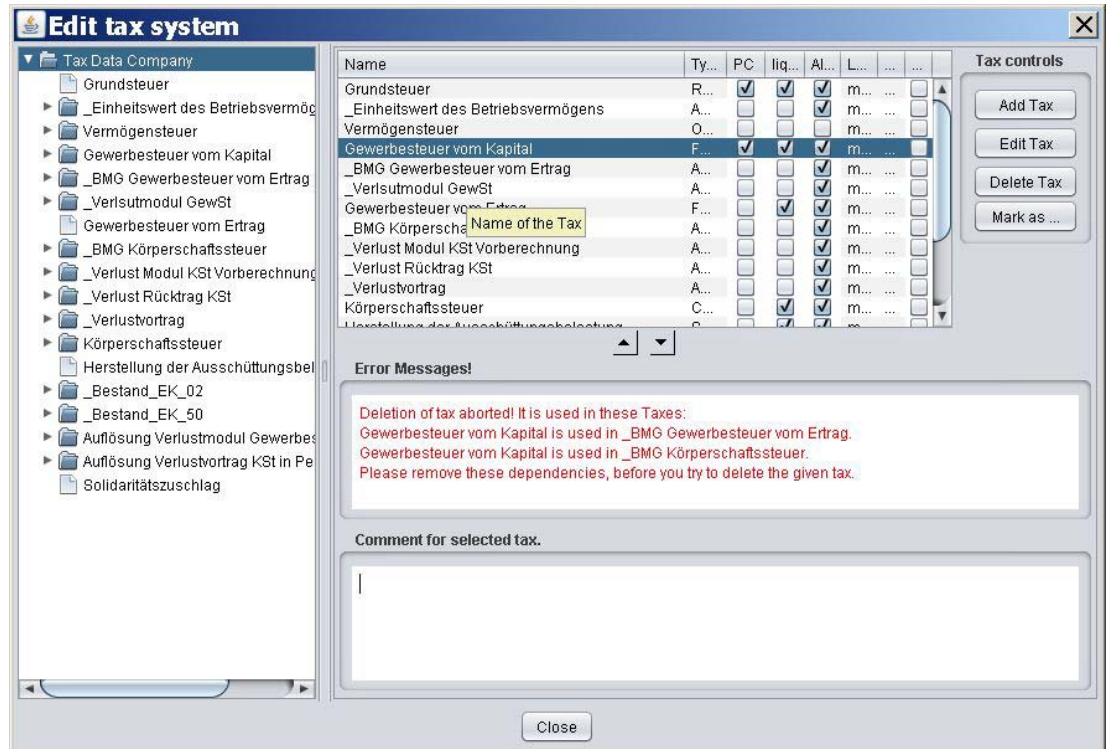
Zur Vermeidung fehlerhafter Eingaben und um einem daraus eventuell resultierenden inkonsistenten Zustand der Daten zu begegnen, sind besondere Vorkehrungen bezüglich der Einhaltung der korrekten Reihenfolge der einzelnen Steuern zu treffen. Soll die aktuelle Periode verwendet werden, muss die Steuer, auf die zurückgegriffen wird, auf jeden Fall vor der aktuellen berechnet werden. Rückgriffe auf Vorperioden sind dagegen unproblematisch, da diese bereits berechnet wurden. In diesen Fällen kann sogar auf dieselbe Steuer zurückgegriffen werden.

Diese Fälle werden automatisch analysiert und überprüft. Bei Fehlern wird im Basisdialog (ähnlich der Meldung in Abbildung 35) eine Fehlermeldung ausgegeben. Diese kann durch das Verschieben von Steuern oder der Veränderung der Periode des Rückgriffs behoben werden. Solange die Fehlermeldung besteht, wird die Berech-



nung fehlerhaft sein. Im schlimmsten Fall gerät das Programm in eine endlose Schleife. Daher sollten solche Fehler immer sofort korrigiert werden.

*Abbildung 35: Fehlermeldung beim Löschen einer Steuer*



Ebenso kann eine Steuer nur gelöscht werden, wenn sie nicht in einer anderen verwendet wird. Das Entfernen wird solange verweigert bis alle Rückgriffe entfernt wurden. Andernfalls würden noch vorhandene Verweise ins Leere gehen und zu undefiniertem Verhalten führen. Auch die Einhaltung der Fremdschlüsselbeziehung in der Datenbank wäre dann nicht mehr garantiert. Für das Auffinden solcher Interdependenzen müssen auch die Steuersysteme der Anteilseigner überprüft werden, da auch dort Rückgriffe auf Steuern der Unternehmensebene möglich sind.

#### 5.3.7.2.3 Verlustverrechnungsmodul

Verluste stellen steuerlich einen Sonderfall dar, da sie in der Regel nicht zu einer direkten Steuergutschrift führen. Dies würde der spiegelbildlichen Anwendung im Vergleich zu Gewinnen entsprechen. Verluste können in der Regel nur vor- oder zurückgetragen werden. Ein besonderes Problem ergibt sich am Ende des Planungshorizontes, da noch bestehende Verluste evtl. in der Zukunft genutzt werden könnten. Für einen abschließenden Vergleich zwischen verschiedenen Ländern müssen diese bewertet werden. Aufgrund unterschiedlicher Beschränkungen, z. B. bzgl. der Zeit,

ist eine einheitliche Erfassung und Bewertung nicht möglich bzw. nicht korrekt. Somit ist die korrekte Verlusterfassung und -verwaltung eine elementare Aufgabe.

In dem bisherigen Metamodell des European Tax Analyzer gab es keine explizite Verlustverrechnung. Vor der Einführung der Bausteine zur Modellierung einer Steuer wurde diese, wie das gesamte Steuersystem eines Landes, auch direkt im Quellcode hinterlegt. Somit führte jegliche Änderung dieser Regelung, wie z. B. der Mindestbesteuerung oder der Verkürzung und Limitierung des Verlustrücktrages zu direktem Programmieraufwand. Dies ist mit dem mittlerweile modularen Aufbau und den erhöhten Anforderungen durch die stark angestiegene Anzahl von Ländern nicht mehr vereinbar.

Auch eine mithilfe der bisher schon vorhandenen Module modellierte Verrechnung ist vor diesem Hintergrund nicht mehr tragbar. Diese kam bisher zur Anwendung. Zum einen ist sie sehr komplex, da für jede Periode ein Art Hilfssteuer angelegt werden muss. Zudem muss jede mögliche intertemporale Interaktion zwischen diesen Konstrukten abgebildet werden. Zum anderen ist diese Methode auch sehr fehleranfällig, da sie in der Regel nur für die aktuell berechneten Modelle validiert werden kann. Weiterhin beschränken sich die Modellierungen, zugunsten einer Reduktion der Komplexität, auf die korrekte Abbildung des konkret vorliegenden Falls. Sie wollen bewusst keine allgemeingültigen Lösungen anbieten. Dies wird nun zum Problem, da durch die Speicherung in der Datenbank die Umsetzungen für alle möglichen Fälle funktionieren müssen. Des Weiteren sind die genauen kodifizierten Regelungen bei den bisherigen Umsetzungen nur sehr schwer zu erkennen, da der mögliche Verfall von Verlusten evtl. erst in der sechsten Hilfssteuer sichtbar wird.

Zur Vereinfachung und Generalisierung der Modellierung wird auf Basis der aktuellen deutschen Rechtslage ein Verlustmodul entwickelt. Zudem sollen die konkreten Regelungen möglichst auf einen Blick sichtbar sein und somit auch die Abbildung in der Datenbank für die Recherche optimiert werden. Des Weiteren soll dieses Modul auch zur Verwaltung von Freibeträgen nutzbar sein, da diese, soweit vortragbar, einen ähnlichen Fall darstellen.

Für die konkrete Umsetzung wurde die bereits vorhandene Infrastruktur dergestalt verwendet, dass die Eingabeparameter bzgl. des Verlustes oder Freibetrages und des verwendbaren Gewinns durch separate Sammelbausteine realisiert wurden. Im einfachsten Fall, falls die Bemessungsgrundlage bereits als eigenes Steuermodul model-

liert wurde, reicht ein entsprechender Verweis.<sup>174</sup> Zur Vereinfachung erwartet das Modul immer positive Werte, auch bei Verlusten. Werte kleiner Null werden als Null interpretiert. Dies ist kein Nachteil, da es kein Problem darstellt, die Ergebnisse vorher im Sammelbaustein zu negieren.<sup>175</sup> Diese Annahme hat den Vorteil, dass nun Verluste (i. d. R. negativ) und Freibeträge (i. d. R. positiv) einheitlich behandelt werden können. Des Weiteren kann nun auf explizite Konstruktionen, die das Maximum zwischen null und dem jeweiligen Resultat ermitteln, verzichtet werden.

Ziel des Moduls ist es, den verwendbaren Verlust bzw. Freibetrag in der aktuellen Periode zu ermitteln. Damit ist es anschließend möglich, die erwartete Steuerrückstattung oder die Minderung der Bemessungsgrundlage zu berechnen. Im Falle eines Verlustrücktrags stellt es kein Problem dar, dass diese Kalkulation erst für die folgende Periode durchgeführt werden kann, da auch in der Realität die liquiditätswirksame Rückerstattung erst im nächsten Veranlagungszeitraum erfolgt.

*Abbildung 36: GUI des Verlustmoduls für den Vortrag*

Hat eine Gesellschaft z. B. in 2008 einen Gewinn erzielt und erleidet in 2009 einen Verlust, wird der maximal mögliche Rücktrag ermittelt und ausgegeben. Die diesen Rücktrag übersteigenden Verluste verbleiben im Modul und werden entsprechend

<sup>174</sup> Dieser wird über das Modul „Andere Steuern“ realisiert.

<sup>175</sup> Diesen Effekt kann man z. B. durch die Multiplikation mit -1 erzielen.

vorgetragen. In einem weiteren Schritt kann anschließend die Höhe der Steuerrück-  
erstattung für 2009 (i. d. R. Körperschaftsteuersatz mal Höhe des Rücktrags - bei li-  
nearen Tarifen) ermittelt werden. Diese Erstattung erhöht dann 2009 die Liquidität  
des Unternehmens. Ebenso erfolgt auch bei komplexeren Sachverhalten die liquidi-  
tätswirksame Verrechnung immer in der Periode, in der ein Rücktrag bzw. Vortrag  
möglich wird.

Tabelle 3 demonstriert die Arbeitsweise des Moduls. Der Vortrag ist auf 60 % des  
eine Million übersteigenden aktuellen Gewinns beschränkt. Die Beschränkung des  
Rücktrags liegt bei 511.500 und ist nur für die vorangegangene Periode möglich.

*Tabelle 3: Verlustverrechnungsbeispiel*

Periode:	1	2	3	4	5	6	7	8	9	10
Verlust (positiv)	50.000.000 €	0 €	0 €	40.000.000 €	0 €	10.000.000 €	0 €	0 €	20.000.000 €	0 €
Gewinn	0 €	30.000.000 €	20.000.000 €	0 €	80.000.000 €	0 €	25.000.000 €	45.000.000 €	0 €	67.000.000 €
Verlustvortrag	50.000.000 €	31.600.000 €	19.200.000 €	58.688.500 €	10.288.500 €	19.777.000 €	4.377.000 €	0 €	19.488.500 €	0 €
Rücktrag	0 €	0 €	0 €	511.500 €	0 €	511.500 €	0 €	0 €	511.500 €	0 €
Verwendbarer Restgewinn für Rücktrag:	0 €	11.600.000 €	7.600.000 €	0 €	31.600.000 €	0 €	9.600.000 €	40.623.000 €	0 €	47.511.500 €
<b>Verwendbarer Verlustvortrag</b>	0 €	18.400.000 €	12.400.000 €	511.500 €	48.400.000 €	511.500 €	15.400.000 €	4.377.000 €	511.500 €	19.488.500 €
Verbleibender Verlustvortrag	50.000.000 €	31.600.000 €	19.200.000 €	58.688.500 €	10.288.500 €	19.777.000 €	4.377.000 €	0 €	19.488.500 €	0 €

In Periode eins entsteht ein Verlust von 50 Millionen € der zum Teil mit dem Ge-  
winn aus Periode zwei verrechnet werden kann. Der maximal mögliche Betrag be-  
trägt 18,4 Millionen € und stellt die Ausgabe des Moduls dar.<sup>176</sup> Somit verbleibt ein  
restlicher Verlustvortrag im Modul in Höhe von 31,6 Millionen €. In Periode drei  
reduziert sich der restliche Verlustvortrag um weitere 12,4 Millionen €. Ein Rücktrag  
ist erstmals in Periode vier möglich. Die folgenden Perioden werden nach dem glei-  
chen Schema berechnet und in Periode 10 sind dann alle Verluste verrechnet.

Alternativ kann man sich auch die restlichen Verluste oder Freibeträge ausgeben las-  
sen. Dies ist insbesondere dann erforderlich, wenn am Ende des Planungshorizonts  
die restlichen Vorträge bewertet werden müssen.

<sup>176</sup> Von den 30 Mill. Gewinn kann eine Million im vollen Umfang berücksichtigt werden. Von den  
restlichen 29 Millionen aber nur 60 % ( $18,4 = 29 \cdot 0,6 + 1$ ).

Abbildung 37: GUI des Verlustmoduls für den Rücktrag

Eine Besonderheit ergibt sich jedoch für den Fall, dass ein progressiver Tarif benutzt wird und ein Rücktrag von mehr als einer Periode erlaubt ist. In diesen Fällen ist es leider nicht mehr möglich zu ermitteln, in welcher Periode die Bemessungsgrundlage um welchen Betrag gemindert worden wäre, um somit die exakte Tarifbelastung für diese Perioden zu ermitteln. Dieser Fall dürfte jedoch äußerst selten vorkommen und konnte bei den aktuellen Rechtsständen der bisher betrachteten Länder<sup>177</sup> nicht beobachtet werden. Beim Vortrag stellt sich diese Problematik nicht, da hier der verwendbare Verlust ja direkt von der aktuellen Bemessungsgrundlage abziehbar ist und dann auch progressive Tarife und Stufentarife korrekt berechnet werden.

Für die Berechnung von Freibeträgen ist es zudem möglich, einen Zinssatz anzugeben, mit dem sich restliche Freibeträge verzinsen. Eine solche Form der Fortschreibung ungenutzter Freibeträge wurde zum Beispiel bei der dualen Einkommensteuer vorgeschlagen.<sup>178</sup>

<sup>177</sup> In Deutschland gab es früher die Möglichkeit, mehr als eine Periode zurückzutragen. In Verbindung mit den unterschiedlichen Steuersätzen, für einbehaltene und ausgeschüttete Gewinne, ergibt sich im Ausschüttungsfall eine ähnliche Problematik. Vgl. Spengel, C., et al., Substanzbesteuerung, 2010, S. 36.

<sup>178</sup> Vgl. Sachverständigenrat/MPI/ZEW, Duale Einkommensteuer, 2006, S. 23 f.

Die restlichen Optionen ermöglichen es, die Begrenzung der Rück- (vgl. Abbildung 37) oder Vorträge (vgl. Abbildung 36) genau zu justieren. In Deutschland ist z. B. der Vortrag auf maximal eine Million Euro plus 60 %<sup>179</sup> des eine Million übersteigenden aktuellen Gewinns begrenzt. Des Weiteren liegt die Begrenzung des Verlustrücktrags bei 511.500 €<sup>180</sup>

Aufgrund der bisherigen Infrastruktur ist es leider nur möglich, pro Modul genau einen Wert zurück zu geben. Somit ist es erforderlich, falls man sowohl den verwendbaren als auch den noch verbleibenden Verlust ermitteln möchte, das Modul zweimal zu verwenden. Dies ist insbesondere am Ende des Planungshorizontes relevant, da hier der noch vorhandene Verlustvortrag i. d. R. bewertet werden muss.

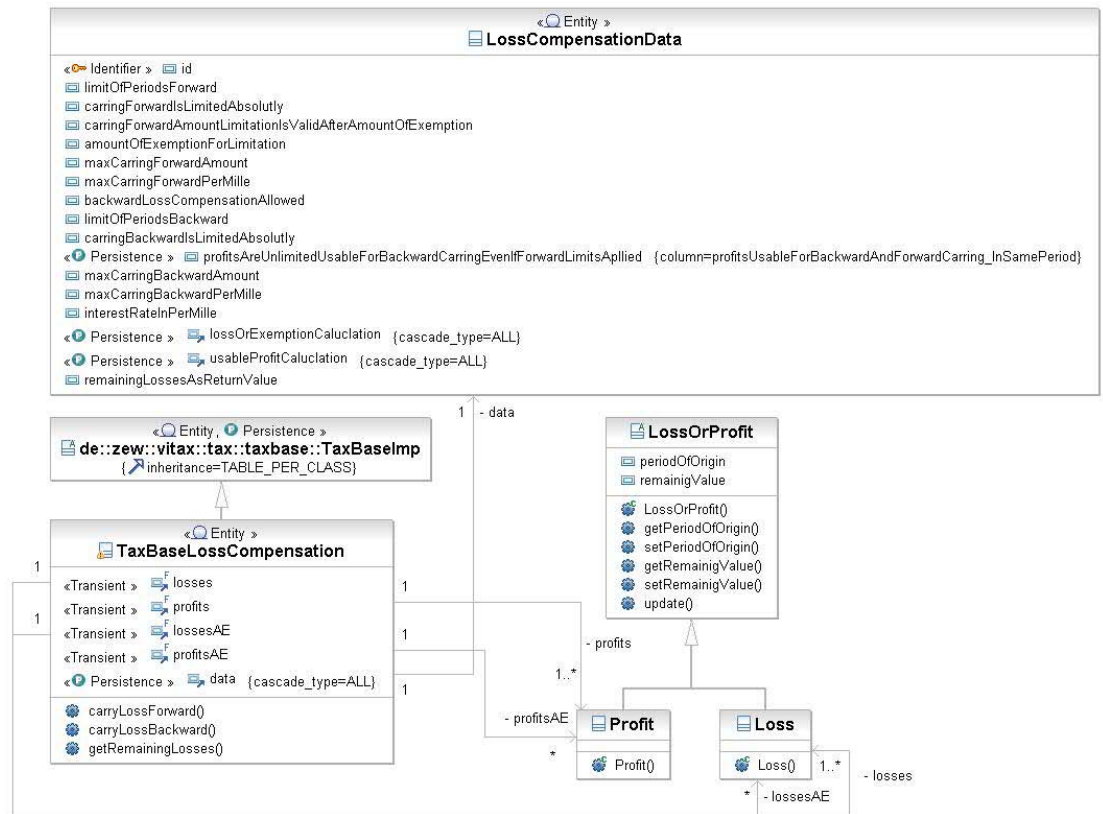
In Abbildung 38 ist das resultierende Klassendiagramm zu sehen. *LossCompensationData* speichert die oben genannten Parameter für die Verlustverechnung. *TaxBaseLossCompensation* verwaltet die Liste mit den verwendbaren Verlusten bzw. Freibeträgen und den Gewinnen.

---

<sup>179</sup> Vgl. § 10d Abs. 2 Satz 1 EStG.

<sup>180</sup> Vgl. § 10d Abs. 1 Satz 1 EStG.

Abbildung 38: Datenmodell und Implementierung der Verlustverrechnung



*Profit* und *Loss* (vgl. Abbildung 38) sind genau genommen keine Spezialisierungen von *LossOrProfit*, da sie das Verhalten bisher nicht ändern oder erweitern.<sup>181</sup> Daher wären sie im Grunde genommen überflüssig. Diese Konstruktion dient jedoch dazu, Fehler zu vermeiden bzw. früher zu entdecken. Bei entsprechender Deklaration<sup>182</sup> der verwaltenden Listen ist es nämlich unmöglich, den Gewinnen einen Verlust hinzu zu fügen und umgekehrt. Eine solche Liste kann dann nur noch Objekte z. B. des Typs *Profit* enthalten. Der Versuch dieser Liste ein Element des Typs *Loss* hinzu zu fügen, würde bereits vom Compiler als Fehler erkannt und zurückgewiesen.

<sup>181</sup> Ein Amphibienfahrzeug kann so z. B. nicht nur wie ein normales Fahrzeug an Land sondern auch im Wasser fahren und stellt somit eine Erweiterung des möglichen Verhaltens bzw. Einsatzgebietes dar.

<sup>182</sup> Seit Java 5 ist es möglich, generische Klassen mit Typ-Parametern zu versehen. Vgl. Mahmoud, Q. H., <http://java.sun.com/.../generics/>, 17. 09. 2009.

#### 5.3.7.2.4 Formelbasierte Bemessungsgrundlagen

##### 5.3.7.2.4.1 Grundlagen

Nach den bereits betrachteten Bemessungsgrundlagen (Sammelbaustein, andere Steuern und Verlustverrechnung) sind die folgenden dadurch gekennzeichnet, dass der Anwender eine Formel hinterlegen kann, die auf bestimmte Werte eines Objektes (z. B. einer Maschine) zurückgreift. Neben der reinen Syntaxprüfung (Validierung) und grundlegenden mathematischen Funktionen<sup>183</sup> wird zudem die Definition von Variablen und deren Auflösung zur Laufzeit benötigt.

Bei der Grundsteuer werden z. B. die Anschaffungskosten der Gebäude und des Grund und Bodens benötigt. Hierfür müsste der Anwender im Metamodell in der Bemessungsgrundlage Bilanz<sup>184</sup> bei den entsprechenden Ausprägungen eine Formel z. B. „ $ahk * 0.2$ “ hinterlegen. Die Variable „ $ahk$ “ muss aber definiert sein und das Programm muss den entsprechenden Eingabeparameter des Modellunternehmens zurückliefern. Die „ $* 0.2$ “ stellt in diesem Beispiel die Modifikation durch mathematische Funktionen dar.

Für die Auflösung der Variablen benutzte die bisherige Lösung Vergleiche von Zeichenketten zur Laufzeit. Dies hat den Nachteil, dass ein Fehler bei dieser Auflösung entweder gar nicht gefunden wurde<sup>185</sup> oder erst zur Laufzeit zu einer Ausnahme geführt hat.<sup>186</sup> In einem ersten Schritt wurde dies dadurch abgemildert, dass der Benutzer die Formel bereits bei der Eingabe prüfen konnte bzw. musste.

Somit sind zwar falsche Benutzereingaben durch nicht definierte Variablen oder fehlerhafte Formeln nahezu ausgeschlossen. Jedoch bleibt die Problematik bestehen, dass eine aufgelöste Variable im aktuellen Kontext nicht gültig sein könnte.

---

<sup>183</sup> Diese Funktionen werden von der Bibliothek JEP in Version 2.4.1 bereitgestellt. Seit Version 3.x ist sie kostenpflichtig. Vgl. Singular Systems, JEP, <http://www.singularsys.com/jep/>, 31. 8. 2009.

<sup>184</sup> Diese enthält auch alle Parameter von Wirtschaftsgütern, die zu den einzelnen Bilanzpositionen gehören und nicht nur die aktuellen Buchwerte.

<sup>185</sup> Der Fall einer nichtdeklarierten oder einer deklarierten, aber nicht aufgelösten Variable wurde meistens nicht explizit behandelt und einfach Null zurückgegeben.

<sup>186</sup> „[...] it pays to discover errors as soon as possible after they are made, ideally at compile time.“ Bloch, J., Effective Java, 2008, S. 110



Deswegen wird eine neue Infrastruktur zur Verwaltung und Auflösung von Variablen eingeführt. Die Vorteile dieser Lösung, die auf Reflection<sup>187</sup> basiert, liegen einerseits darin, dass Fehler bereits während der Übersetzung oder direkt nach dem Start des Programms gefunden werden. Andererseits können die Variablen vererbt werden. Somit muss jede Klasse nur zusätzliche Variable definieren, die nicht bereits von der Oberklasse definiert wurden. Sie kann jedoch den Variablen durch Redefinition auch eine andere semantische Bedeutung zuweisen. Dies ist z. B. bei der Beteiligung notwendig, die von der Klasse Asset abgeleitet wurde, da der aktuelle Wert nun nicht mehr nur aus den fortgeschriebenen Anschaffungskosten besteht. Dieser muss auch die bereits angesparten Gewinnrücklagen berücksichtigen.

Die neue Struktur reduziert auch die Fehlerwahrscheinlichkeit deutlich, da nun die redundante Definition von gleichen Variablen innerhalb einer Klassenhierarchie entfällt. Des Weiteren werden so Namenskonflikte frühzeitig bemerkt und verhindert, da doppelte Variablen aufgrund der Definition der Gleichheit und der Set-Eigenschaften<sup>188</sup> nicht gleichzeitig vorkommen können.

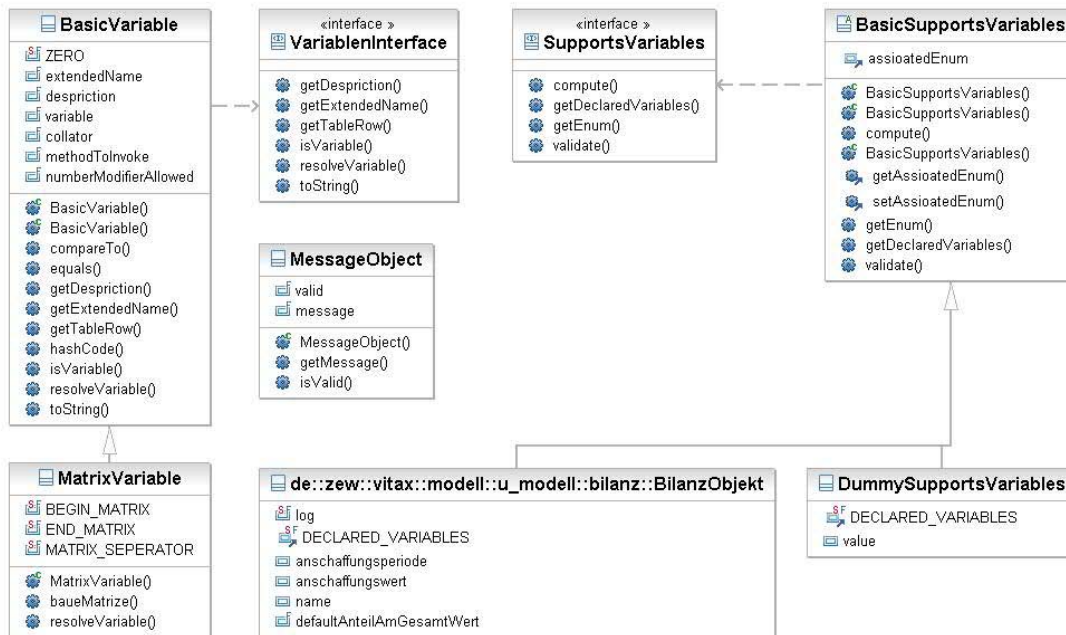
Weiterhin wurde die Anordnung der Elemente (z. B. Maschinen) in der Baumstruktur (vgl. Abbildung 42) überarbeitet und gestrafft. Somit entfallen nun viele Knoten, die bisher nur ein Blatt enthielten. Zudem erleichtert die alphabetisch sortierte Anzeige die Navigation.

---

<sup>187</sup> Mittels dieser Technik ist es möglich, zur Laufzeit eines Programms Informationen (z. B. Methoden und Konstruktoren) über eine Klasse zu ermitteln und aufzurufen. Daher kann der Methodenname als Konstruktorargument der Variable übergeben werden. Durch die Erzeugung aller Variablen bereits beim Start führen fehlerhafte Auflösungen sofort zu Ausnahmen und können somit direkt korrigiert werden.

<sup>188</sup> Ein Set erlaubt keine doppelten Elemente. Der Versuch ein bereits enthaltenes Element erneut hinzuzufügen wird abgelehnt. Der Gleichheitsvergleich wurde auf den Variablennamen beschränkt, so dass die genaue Semantik der Variable keine Rolle spielt. Eine Redefinition einer geerbten Variablen erfordert somit das vorherige explizite Entfernen der alten Variable aus dem Set.

Abbildung 39: Infrastruktur zur Auflösung von Variablen



In der obigen Abbildung 39 sieht man auf der linken Seite die Klassenhierarchie zum Erzeugen von Variablen. Bei der *MatrixVariablen* handelt es sich um einen Spezialfall, da dieser Typ vor der Berechnung noch in eine separate Teilformel umgewandelt werden muss. Soll z. B. die Summe aller bisherigen Ausschüttungen berechnet werden, wird die *MatrixVariable* „ausReihe“ durch einen Vektor mit den Werten einzelner Perioden ersetzt. Für die Berechnung der Summe muss dieser Vektor in der ursprünglichen Formel noch mit einem entsprechenden Operator („vsum“) verknüpft sein. Die vollständige Formel lautet „vsum(ausReihe)“ und wird in einem Zwischenschritt z. B. in „vsum([1, 2, 3])“ umgewandelt. Danach wird dieser Ausdruck ausgewertet und im konkreten Beispiel sechs zurückgegeben. Der Typ *MatrixVariable* dient somit der Abkürzung und Vereinfachung von Formeln. Das *VariablenInterface* und *SupportsVariables* könnten bei der jetzigen Implementierung genau genommen entfallen, da sie zurzeit nur von jeweils einer Vererbungshierarchie implementiert werden. Um zukünftige Erweiterungen zu erleichtern, erscheint ein Fortbestand dennoch sinnvoll.

Die Methode *resolveVariable* dient der Evaluierung des Wertes der Variable zur Laufzeit. *IsVariable* dient der Überprüfung, ob die übergebene Zeichenkette der Variable entspricht oder nicht. Die restlichen Methoden werden primär für die grafischen Benutzerdialoge verwendet. Das *MessageObject* dient der uniformen Fehler-

behandlung und enthält im Falle der Inkorrektheit die entsprechende Beschreibung zur Interaktion mit dem Benutzer.

Auf der rechten Seite sieht man die Infrastruktur zur Verwaltung der Variablen. *DummySupportsVariables* dient der Abbildung von Gliederungselementen (z. B. Aktiva), die keine direkte Entsprechung im Unternehmensmodell besitzen und stellt nur die Variable „x“ zur Verfügung,<sup>189</sup> die es ermöglicht, auf das Ergebnis der Unterelemente zuzugreifen. Aus Gründen der Übersichtlichkeit ist hier mit *BilanzObjekt* nur eine weitere Spezialisierung von *BasicSupportsVariables* aufgeführt. Die konkrete Implementierung dieser Infrastruktur verwaltet die Liste aller möglichen Variablen in einem *SortedSet*.<sup>190</sup> Dieses kann durch die Unterklasse kopiert<sup>191</sup> und entsprechend erweitert werden. Jede Instanz einer Klasse, die das Interface implementiert oder die abstrakte Klasse erweitert, muss auch eine Instanz einer Aufzählung (vgl. Abbildung 16) bereithalten, um die Rekonstruktion der Verknüpfung zwischen dem ökonomischen Modell und den steuerlichen Daten zu gewährleisten.

Diese Infrastruktur zur Auflösung von Variablen enthält zwar keine expliziten Informationen, die in der Datenbank zu speichern wären. Die Stabilität dieses Systems ist jedoch enorm wichtig, da die hinterlegten Formeln diese Variablen enthalten.

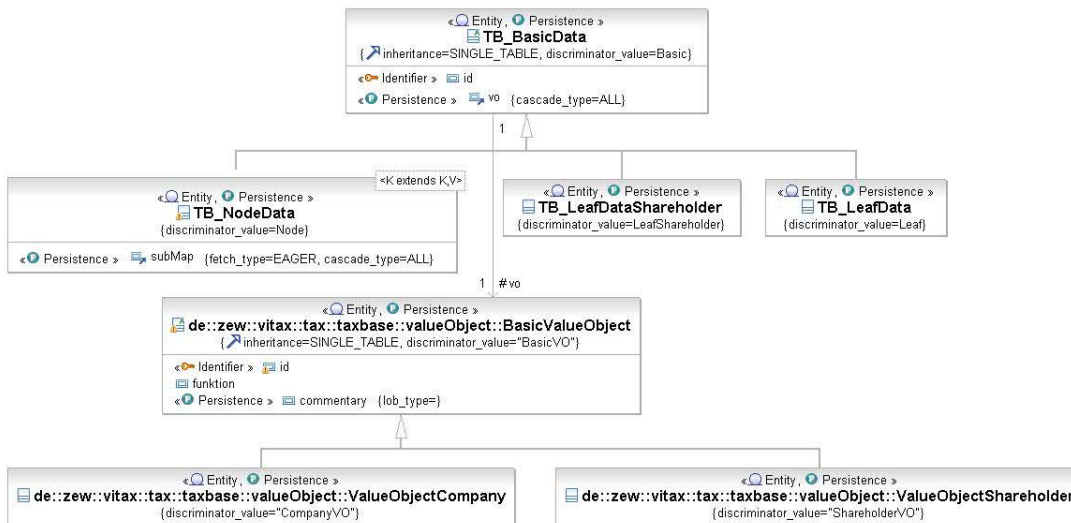
---

<sup>189</sup> Dadurch können auf einfache Weise ganze Unterbäume von der Berechnung ausgeschlossen werden, ohne jede einzelne Formel löschen zu müssen. Des Weiteren können auf dieser Ebene Modifikationen erfolgen, die alle Unterknoten betreffen.

<sup>190</sup> *Equals* und *hashCode* wurden in *BasicVariable* entsprechend überschrieben. Auch bei einer Implementierung des Interfaces sollte darauf geachtet werden.

<sup>191</sup> Eine Kopie der Referenzen auf die Objekte reicht hierbei vollkommen aus, da *BasicVariable* das Immutable Pattern implementiert.

**Abbildung 40: Infrastruktur zur Unterstützung der Baumstruktur der formelbasierten Bemessungsgrundlagen**



Zur einheitlichen Abbildung der Gliederung der Bilanz (z. B. Aktiva → Wirtschaftsgüter → Gebäude) und der Gewinn-und-Verlust-Rechnung (Betriebsergebnis → Material) wurden Gliederungsklassen (vgl. Abbildung 40) angelegt. Für jede Gliederungsebene wird ein Element vom Typ *TB\_NodeData* eingefügt, das eine Map (assoziatives Array) enthält. Als Schlüsselement wird eine der Aufzählungsklassen aus Abbildung 15 verwendet. Konkrete Instanzen der Unterklassen von *TB\_Basic\_Data* bilden die Werte der Schlüssel-Wert-Beziehung. Ist keine weitere Untergliederung gewünscht, kommen *TB\_LeafData* und *TB\_LeafDataShareholder* zum Einsatz.

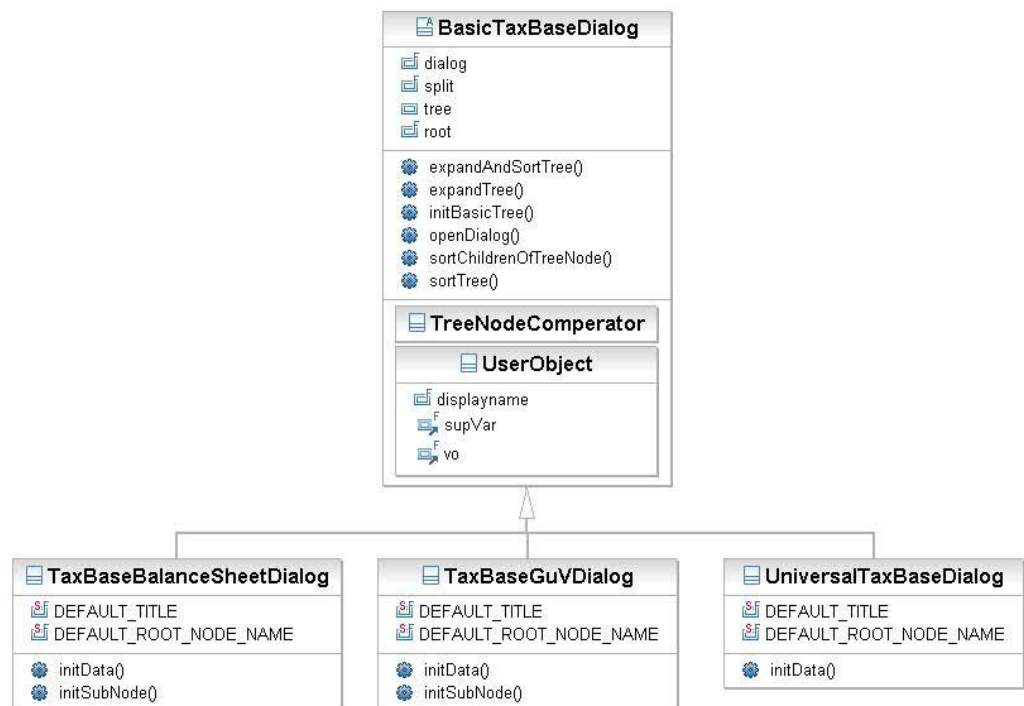
Diese unterscheiden sich nur durch die Verwendung einer anderen Spezialisierung des *BasicValueObject*. Dieses speichert die hinterlegte Formel. Die Zuordnung zu den konkreten Instanzen des Unternehmensmodells erfolgt über die Auflösung der Schlüsselattribute der assoziativen Arrays. *ValueObjectCompany* und *ValueObjectShareholder* stellen für die Unternehmens- und Anteilseignerebene unterschiedliche Berechnungsmethoden zur Verfügung. Sie haben jedoch keinerlei zusätzlichen Attribute, sodass alle Instanzen in einer Tabelle (Single Table) verwaltet werden können. Die ebenfalls von *BasicValueObject* abgeleitete Klasse *ValueProductionCosts* (vgl. Abbildung 18) enthält nur zwei boolean<sup>192</sup> Attribute, sodass der zusätzliche Speicherplatzverbrauch durch nicht genutzte Attribute sehr gering ist und zu Gunsten einer einfacheren Struktur in Kauf genommen wird. Ebenso werden alle Spezialisie-

<sup>192</sup> Für die Abbildung von wahr oder falsch wird jeweils nur ein Bit benötigt.

rungen von *TB\_Basic\_Data* abgebildet, da die *submap* von *TB\_NodeData* über eine Fremdschlüsselbeziehung in einer Verknüpfungstabelle abgebildet wird und somit kein zusätzliches Attribut benötigt wird.

Die Bemessungsgrundlage der Anteilseigner benötigt diese Baumstruktur nicht zwingend. Sie benutzen diese Infrastruktur aber trotzdem, da damit die Abbildung einheitlicher ist und keine zusätzlichen Tabellen in der Datenbank benötigt werden. Ebenso können dadurch die für diese Struktur bereitgestellten Dialoge (vgl. Abbildung 41) wiederverwendet werden.

**Abbildung 41: Dialog-Hierarchie für formelbasierte Bemessungsgrundlagen**



Bei der Überarbeitung wurde darauf geachtet, dass alle Bemessungsgrundlagen, die diese Abstraktion benötigen, die gleiche Implementierung nutzen können und somit auch durch denselben Dialog darstellbar sind. Dadurch wurde das mögliche Fehlerpotenzial stark reduziert, da es nun nur noch eine konkrete Implementierung anstelle der bisherigen vier gibt.

Der *BasicTaxBaseDialog* (vgl. Abbildung 41) stellt den Dialog<sup>193</sup> an sich dar. Das *UserObject* hält die Informationen für die einzelnen Blattknoten des Auswahlbaumes

<sup>193</sup> Vgl. z. B. Abbildung 43.

bereit. Somit müssen die Unterklassen nur noch diesen Baum entsprechend initialisieren. Die Referenz *supVar* (vgl. Abbildung 39 - *SupportVariables*) verweist auf Objekte, die Variablen unterstützen. Mithilfe der Methode *getTableRow()* von *BasicVariable*<sup>194</sup> wird dann dynamisch, bei Auswahl des entsprechenden Knotens, die Tabellenzeile erzeugt. Diese enthält den Variablennamen und eine Kurzbeschreibung. Anschließend wird der Tabelleninhalt des Dialogs aktualisiert.

*TaxBaseBalanceSheetDialog* stellt die Spezialisierung für die Bilanz (vgl. Abbildung 42) dar. Entsprechend repräsentiert *TaxBaseGuVDialog* (vgl. Abbildung 43) diese für die Gewinn-und-Verlust-Rechnung. *UniversalTaxBaseDialog* wird für die Einkunftsarten (vgl. Abbildung 44) und Vermögensgegenstände (vgl. Abbildung 45) der Anteilseigner verwendet.

#### 5.3.7.2.4.2 Bilanz

Für die Ermittlung der Bemessungsgrundlage einer Steuer ist es häufig notwendig, auf bilanzielle Größen zurückzugreifen. Hiermit sind nicht nur die Parameter gemeint, die zur Darstellung der Bilanz zwingend notwendig sind, sondern auch alle Werte, die für die einzelnen Wirtschaftsgüter, Verbindlichkeiten und andere Positionen hinterlegt wurden. Bei der Grundsteuer werden z. B. die Anschaffungs- oder Herstellungskosten<sup>195</sup> eines Gebäudes benötigt. Dieser Rückgriff wird durch die Bereitstellung von Variablen für die zu hinterlegende Formel realisiert.<sup>196</sup> Dieses Prinzip ist ein elementares und wird von allen formelbasierten Bemessungsgrundlagen realisiert.

Durch die Reduktion der in der Datenbank gespeicherten Informationen auf rein steuerlich relevante Informationen ergibt sich die Problematik, dass diese Verknüpfung zwischen dem ökonomischen Modell des Unternehmens<sup>197</sup> und den Vorschriften zur Ermittlung der Steuerlast verloren geht.

---

<sup>194</sup> Siehe Abbildung 39.

<sup>195</sup> In Estland z. B. ist dies der Fall. In Deutschland stellt sich die Ermittlung des Einheitswertes etwas komplizierter dar.

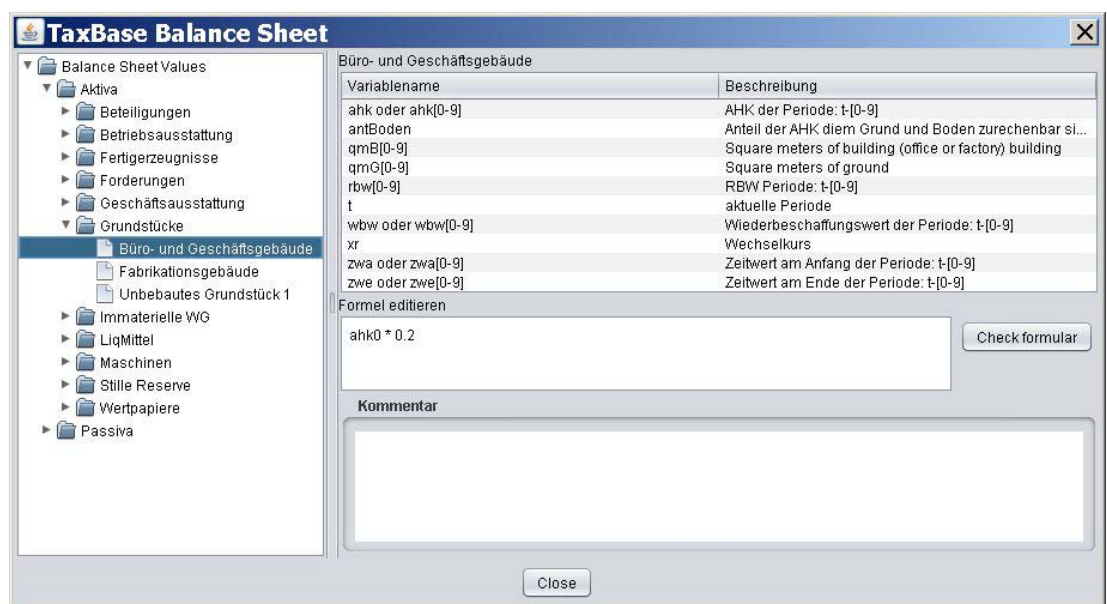
<sup>196</sup> Vgl. Stetter, T., VITAX, 2005, S. 78 ff.

<sup>197</sup> Auch die Steuerbilanz an sich stellt in diesem Sinne keine steuerlich relevante Information dar. Sie bildet im Grunde nur das Ergebnis der Anwendung der jeweiligen Vorschriften ab und kann somit jederzeit reproduziert werden.

Deswegen ist es notwendig, eine Schnittstelle zu definieren, die es erlaubt, diese Referenzen beim Laden der Steuerinformation zu rekonstruieren. Die bisher implizit verwendete Lösung, dieses Problem über konkrete Objektreferenzen zu realisieren, ist nicht mehr möglich, da diese nur zur Laufzeit des Programms eindeutig sind. Bei der bisherigen Speicherung in XML wurden diese temporären Referenzen in Verweise umgewandelt, die innerhalb des Dokumentes eindeutig waren. Zudem enthielt das Dokument auch die referenzierten Objekte. Bei der Verwendung eines anderen ökonomischen Modells mussten diese Referenzen quasi „von Hand“ ersetzt werden.

Dieses Problem des Ersetzens würde auch bestehen, falls man jedem Objekt eine synthetische ID zuweisen würde, da diese nur für das aktuelle Objekt gültig wäre und nicht ohne Weiteres auf ähnliche Objekte übertragbar ist. Diese Übertragbarkeit ist jedoch eine grundlegende Voraussetzung, um die steuerlichen Informationen mit möglichst vielen verschiedenen ökonomischen Modellen (z. B. Branchen) berechnen zu können.

**Abbildung 42: Benutzeroberfläche für die Bemessungsgrundlage Bilanz**



Somit erscheint es am sinnvollsten, die direkte Verknüpfung durch eine indirekte zu ersetzen. Indirekt bedeutet hierbei, dass nicht das konkrete Objekt Ziel der Verknüpfung ist, sondern nur der modelltheoretische Platzhalter. So soll vereinfacht ausgedrückt z. B. nur gespeichert werden, dass die Anschaffungskosten von „Maschine 1“ zur Berechnung benötigt werden und nicht auf deren konkrete Ausprägung (Nutzungsdauer: 5 Jahre, Anschaffungskosten: 2 Mill. € usw.) verweisen. Die Verbin-

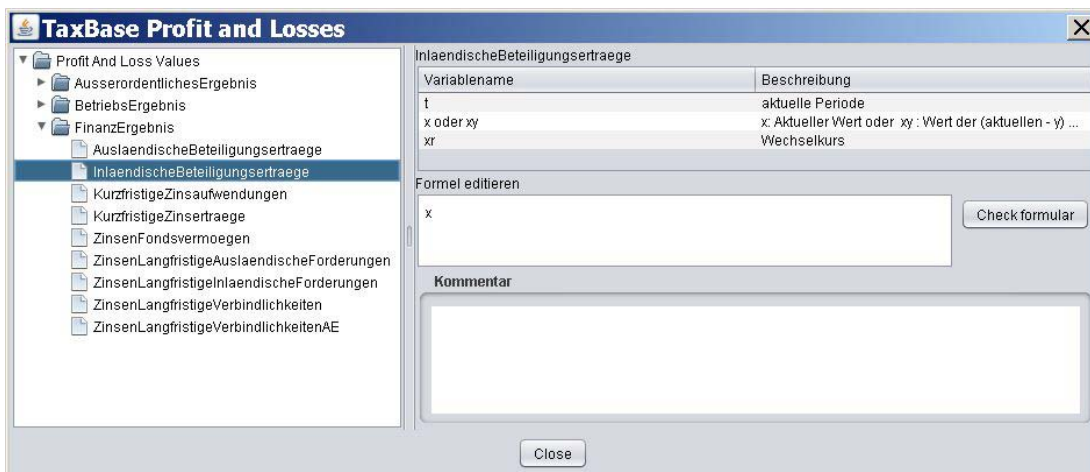
dung wird dann zur Laufzeit des Programms dynamisch rekonstruiert, indem die indirekte Referenz durch einen direkten Verweis auf das konkrete Objekt ersetzt wird.

In Java wurde dies durch verschiedene Aufzählungstypen realisiert. Diese werden in der Datenbank durch Zeichenketten abgebildet, um bei nachträglichen Erweiterungen nicht an eine konkrete Reihenfolge gebunden zu sein, was bei einer ordinalen Abbildung der Fall gewesen wäre.<sup>198</sup>

#### 5.3.7.2.4.3 Gewinn-und-Verlust-Rechnung

Ein Rückgriff auf die Parameter der Gewinn-und-Verlust-Rechnung via Formeln, ähnlich der Bilanz, ist für die Ermittlung der Bemessungsgrundlagen ebenso notwendig. Der primäre Unterschied liegt darin, dass es sich um reine Ergebnisgrößen handelt. Hierbei wird zwischen dem außerordentlichen, dem Betriebs- und dem Finanzergebnis unterschieden. Ein weiterer Unterschied besteht darin, dass die Vielfalt der Variablen sehr begrenzt ist, da primär für jede Größe nur der aktuelle Wert oder die Daten aus den Vorperioden vorliegen. Bei Aufwandsposten, die auch in die Herstellungskostenermittlung (z. B. beim Personalaufwand) einfließen können, kann i. d. R. auch auf die ursprünglichen Gesamtgrößen<sup>199</sup> zurückgegriffen werden.

Abbildung 43: Benutzeroberfläche für die Gewinn-und-Verlust-Rechnung



<sup>198</sup> Es wäre wünschenswert gewesen, diese Strukturen auch durch *java.util.EnumMap*'s zu realisieren. Diese sind aber nicht mit dem *java.beans.XMLEncoder* kompatibel, da ein argumentloser Konstruktor fehlt und der für das notwendige Attribut zur Implementierung eines *PersistenceDelegate* nicht die notwendigen Zugriffsmethoden bereitstellt. Somit wurde auf *java.util.TreeMap* ausgewichen.

<sup>199</sup> Diese sind normalerweise nicht Bestandteil der Gewinn-und-Verlust-Rechnung nach dem Umsatzkostenverfahren, werden aber trotzdem in dieser Bemessungsgrundlage bereit gestellt.



#### 5.3.7.2.4.4 Anteilseigner

Die Berücksichtigung des Anteilseigners ist für eine reine Bewertung einer Unternehmenssteuer, die auf die Besteuerung einer Körperschaft abzielt, nicht zwingend erforderlich.<sup>200</sup> Bei transparenten Besteuerungsformen, wie z. B. der Personengesellschaft, oder bei der Betrachtung der Gesamtsteuerbelastung einer Kapitalgesellschaft unter Einbezug der Anteilseigner ist dieser jedoch unentbehrlich.

Anteilseigner werden als natürliche Personen i. d. R. progressiv besteuert, d. h. der Grenzsteuersatz steigt mit steigendem Gesamteinkommen. Somit spiegelt eine isolierte Betrachtung der Rückflüsse aus der Kapitalgesellschaft über Dividenden oder – im Fall von Gesellschafterdarlehn – Zinsen evtl. nicht die tatsächliche Steuerbelastung wider. Daher müssen auch andere Einkunftsarten simuliert werden.

Zudem kann die Ebene der Anteilseigner bei einer transparenten Besteuerung von besonderem Interesse sein, da nicht alle Wirtschaftsgüter eines Betriebes auch in dessen Eigentum stehen müssen. Hierbei sei insbesondere auf die Möglichkeit der Vermietung und Verpachtung von unbebauten Grundstücken oder Gebäuden inkl. Grund und Boden verwiesen. Des Weiteren ist auch die Vergütung eines Anteilseigners als Geschäftsführer besonders zu beachten.

Um sich ein Gesamtbild der vorliegenden steuerlichen Situation zu verschaffen, ist die Modellierung von verschiedenen Vermögensgegenständen und Einkunftsarten notwendig. Ebenso ist es auch auf der Ebene des Anteilseigners erforderlich, Zugriff auf alle Variablen des Unternehmensmodells zu haben.

---

<sup>200</sup> Dies ist nur der Fall, wenn man konstante Ausschüttungen unterstellt. Z. B. würden prozentuale Ausschüttungen des Jahresüberschusses den Endvermögenswert des Unternehmens beeinflussen, da die Höhe des Jahresüberschusses direkt von den steuerlichen Regelungen abhängt. Somit liefert dann nur eine Betrachtung der Gesamtebene zwischen den Ländern vergleichbare Ergebnisse.

Abbildung 44: Eingabemaske Anteilseigner Vermögensgegenstände

Variablenname	Beschreibung
abschreibAkt	Abschreibung (komplett) in aktueller Periode
ahk	Anschaffungskosten
anteilGeb	Anteil am Gebäude(Ganzzahl)
egw	Eingabewert
er	Ertrag
mieteAkt	Miete in aktueller Periode
mieteAnf	Miete am Anfang (Periode 1)
t	aktuelle Periode
xr	Wechselkurs
zwa	Inflationierter Wert am Anfang der Periode
zwe	Inflationierter Wert am Ende der Periode

Formel editieren

mieteAkt

Check formular

Kommentar

Close

Wie bereits in 5.3.7.2.4 erwähnt, wird auch hier mittels der Aufzählungstypen *EnumOfAssets* und *IncomeEnum* die Trennung zwischen den steuerlichen Daten und dem ökonomischen Modell realisiert.

Abbildung 45: Eingabemaske Anteilseigner Einkunftsarten

Variablenname	Beschreibung
ahk	Anschaffungskosten
egw	Eingabewert
er	Ertrag
t	aktuelle Periode
xr	Wechselkurs
zwa	Inflationierter Wert am Anfang der Periode
zwe	Inflationierter Wert am Ende der Periode

Formel editieren

zwe

Check formular

Kommentar

Close

Eine Besonderheit stellt die Liquidität des Anteilseigners dar. Sie wird für die Zwecke der Steuermodellierung als Vermögensgegenstand betrachtet. Jedoch muss sie separat berechnet werden, da sie unmittelbar von den Ergebnissen der anderen Vermögensgegenstände und Einkunftsarten sowie der Steuerzahlungen abhängt. Deswe-

gen taucht sie nicht in der entsprechenden Aufzählung auf (vgl. Abbildung 15), um einfaches Iterieren zu gewährleisten und diesen Sonderstatus zu betonen.

#### 5.4 PDF-Verwaltung

Ein zentrales Ziel des Projektes war die Verbesserung der Dokumentation und die zentralisierte Speicherung. Hierfür wurde das PDF-Format, das bereits ein ISO-Standard<sup>201</sup> für die Langzeitarchivierung ist, ausgewählt. Damit sollte auch in naher Zukunft eine Verfügbarkeit der hinterlegten Informationen gewährleistet sein.

Um sicher zu stellen, dass die benötigten Informationen möglichst an der jeweils passenden Stelle hinterlegt werden können, war es wichtig, eine passende Granularität zu finden. Im Extremfall könnte es dem Anwender ermöglicht werden, in jeder Eingabemaske ein solches Dokument zu speichern. Dies würde den Benutzer wahrscheinlich eher irritieren und würde zudem an den bestehenden Eingabemasken und dem Datenmodell viele Änderungen nach sich ziehen.

Daher wurde ein Kompromiss gefunden, der der Tatsache Rechnung trägt, dass primär die eigentliche Steuermodellierung kommentierungs- und dokumentationsbedürftig ist. Somit wurde der Tarif-Dialog ausgewählt, da dieser Teil des Sammelbausteins ist und auf jeder Ebene einer Steuermodellierung vorhanden ist. Dadurch ist eine direkte Verknüpfung zwischen der Modellierung und deren Dokumentation gegeben.

Dies wird vor allem dann von Vorteil sein, wenn sich die Dokumentation an diesen Stellen auf das Wesentlichste konzentriert, da auf diese Weise die benötigten Informationen schneller gefunden werden können. Des Weiteren können auf Länderebene weitere Dokumente hinzugefügt werden. Dies ist z. B. für Abschreibungsregelungen oder Vorschriften zur Ermittlung der Herstellungskosten, die nicht direkter Bestandteil<sup>202</sup> des Metamodells sind, sinnvoll. Ebenso kann von dieser Ebene aus auch auf alle modular hinterlegten Informationen zugegriffen werden, da diese Dokumente beiden Listen hinzugefügt werden.

---

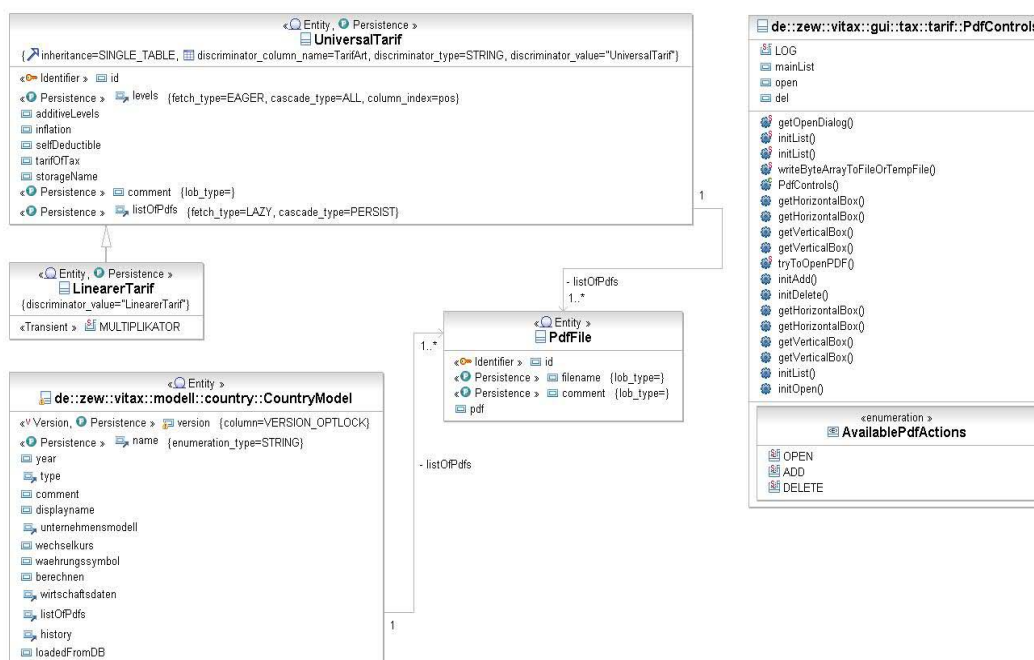
<sup>201</sup> Vgl. Trinkwalder, A., PDF/A als ISO-Standard, <http://www.heise.de/.../63957>.

<sup>202</sup> Diese haben jedoch Einfluss auf die Werte, die über die Bemessungsgrundlagen Bilanz und der Gewinn-und-Verlust-Rechnung verfügbar sind. Abschreibungen wirken sich z. B. auf die Restbuchwerte aus und die Herstellungskostenermittlung beeinflusst direkt die Höhe des Gewinns.

Abbildung 46 verdeutlicht die Beziehungen zwischen den Tarifen (*UniversalTarif* und *LinearerTarif*) und dem Ländermodell (*CountryModel*). *PDFControls* stellt die Kontrollelemente für die grafische Benutzeroberfläche bereit.

Im *PdfFile* werden der Name, ein Kommentar<sup>203</sup> und der Inhalt der Datei als byte-Array hinterlegt. Da PostgreSQL nicht ohne weiteres Large Binary Objects (BLOB) unterstützt, wird das eigentliche Dokument als Tabellenspalte mit Hilfe der Annotation `@Type(type = "org.hibernate.type.BinaryType")` hinterlegt.<sup>204</sup> Normalerweise würde nur eine Referenz auf das BLOB hinterlegt werden. Somit würde die Größe der Spalte und damit auch der Tabelle sehr klein bleiben. In der Datenbank müssen dann Routinen hinterlegt werden, die das Löschen und Aktualisieren dieser Tabelle abfangen und die referenzierten Objekte löschen. Diese Option wurde aufgrund des zusätzlichen Aufwandes und des erhöhten Fehlerpotentials<sup>205</sup> verworfen. Zudem würde eine solche Lösung die Unabhängigkeit<sup>206</sup> von einer konkreten Datenbank konterkarieren.

Abbildung 46: PDF-Verwaltung



<sup>203</sup> Dieser wird zurzeit noch nicht explizit genutzt.

<sup>204</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 134 ff.

<sup>205</sup> Sollten diese Routinen nicht korrekt arbeiten, könnte aufgrund von nicht gelöschten Objekten die Datenbank unbrauchbar werden.

<sup>206</sup> Durch die Wahl von Hibernate genügt es, den Treiber und den zu verwendenden SQL-Dialekt zu spezifizieren. Alle anderen Details sind unabhängig von der verwendeten Datenbank.

Solche Dokumente können leicht mehrere Megabyte groß sein, sodass die gesamte Tabelle bei der gewählten Option recht groß werden kann. Daher werden alle Beziehungen zu *PdfFile* erst bei Bedarf dynamisch nachgeladen. Auf die Möglichkeit, einzelne Attribute dynamisch nachzuladen,<sup>207</sup> wurde verzichtet, da im Allgemeinen davon abgeraten wird und es einen größeren Aufwand erfordert.<sup>208</sup> Zudem sind der Dateiname und der Kommentar als sonstige Attribute von ihrer Größe her unbedeutend, sodass es fast keinen Unterschied macht, ob die ganze Klasse oder nur das Attribut dynamisch nachgeladen wird.

Für die seltenen Fälle, in denen z. B. nur der Dateiname geändert werden soll, wurde die Annotation `@org.hibernate.annotations.Entity(dynamicUpdate = true)`<sup>209</sup> verwendet. Diese bewirkt, dass nur die benötigten Spalten gelesen werden und somit solche Änderungen schnell durchgeführt werden können.

## 5.5 Qualitäts- und Benutzerverwaltung

Einen weiteren zentralen Aspekt stellt die Qualitätssicherung dar. Hierfür ist es unerlässlich, Änderungen am Datenbestand zu protokollieren und deren Güte zu erfassen. Dafür ist es nicht wichtig zu wissen, in welcher Version sich der Datenbestand befindet, da diese nichts über die Aktualität und dessen Qualität aussagt; auch wenn ein loser kausaler Zusammenhang zwischen dem Datum der letzten Bearbeitung und der Aktualität bestehen mag. Dies wird insbesondere dann entscheidend, wenn verschiedene Mitarbeiter denselben Datensatz bearbeiten.

Daher ist es zum Zwecke der Qualitätssicherung erforderlich, auch eine Benutzerverwaltung dergestalt einzuführen, dass auch der Urheber einer Änderung bekannt ist. Des Weiteren kann so auch jedem Benutzer eine Rolle zugeteilt werden. Diese legt fest, welche Änderungen er vornehmen darf. Es erscheint z. B. sinnvoll, nicht allen Benutzern zu erlauben, eine Modellierung eines Landes als geprüft zu markieren.

---

<sup>207</sup> Hierfür wäre eine Bytecode Instrumentalisierung erforderlich. Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 125 ff.

<sup>208</sup> Vgl. O. V., Hibernate Reference Manual, S. 185 ff.

<sup>209</sup> Vgl. Hennebrüder, S., Hibernate Praxisbuch, 2007, S. 320.

Nun stellt sich wiederum die Frage nach der Granularität einer solchen Maßnahme. Reicht es aus, die Modifikationen nur auf Länderebene zu protokollieren oder nicht? Im Idealfall würden, bei der Speicherung der Länder, die Änderungen an den Steuerobjekten automatisch mit protokolliert. Dies ist jedoch nicht umsetzbar, da weder eine manuelle noch eine automatische Überprüfung möglich ist. Die erste Lösung scheidet aus, da durch die Komplexität der möglichen Änderungen eine lückenlose Erfassung mit vertretbarem Aufwand nicht realisierbar erscheint. Die automatische Lösung<sup>210</sup> erkennt zwar geänderte Objekte, darf aber zu diesem Zeitpunkt keine gravierenden Änderungen mehr vornehmen. In diesem Fall betrifft dies das Einfügen eines Elements in die Versionsgeschichte. Des Weiteren hat diese Lösung den Nachteil, dass nur ein Kommentar pro Änderung am jeweiligen Land möglich wäre und somit auch die direkte Zuordnung verloren gehen würde. Um es zu ermöglichen, die Aufgabe der Überprüfung eines Landes in einzelne Teilschritte zerlegen zu können, ist es zudem erforderlich, auch einzelne Steuern validieren zu können.

Nun könnte man, wie bei den Ländern, auch die einzelnen Steuern separat speichern. Dies hätte jedoch zur Folge, dass nun ein Land unverändert<sup>211</sup> erscheint, obwohl Teile des Steuersystems bereits bearbeitet wurden. Dies ist aber nicht gewollt. Zudem könnte sich das nur bei einer stark steigenden Anzahl gleichzeitiger Bearbeitungen, und einer damit einhergehenden Zunahme an Datenverlusten durch Kollisionen (vgl. 5.1.2) in Zukunft eventuell als sinnvoll erweisen.

Die automatische Erfassung aller möglichen Änderungen erscheint daher nicht möglich, sodass den Benutzern die zusätzliche Möglichkeit gegeben wurde, solche Einträge explizit selbst zu erzeugen (vgl. Abbildung 28 „Mark as ...“-Button). Hiermit kann der Anwender eine Änderung oder Überprüfung explizit vermerken bzw. markieren und entsprechend kommentieren.

*ModificationHistoryElement* (vgl. Abbildung 47) stellt einen Eintrag in der Historie von Veränderungen dar. *QualityOfEnteredData* definiert die möglichen Qualitätsstufen („neu“, „kopiert“, „geändert“ und „überprüft“). Die möglichen Rollen der einzelnen Nutzer legt *DefinedRoles* fest.

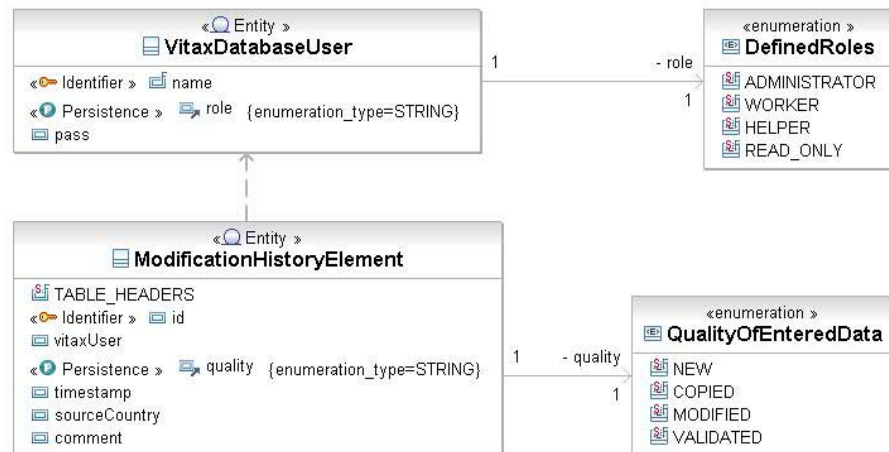
---

<sup>210</sup> Via `org.hibernate.Interceptor`.

<sup>211</sup> Das Versionsattribut würde nicht erhöht werden.

Im *ModificationHistoryElement* wird bewusst keine Referenz auf den *VitaxDatabaseUser* gespeichert. Es wird lediglich der Name gespeichert, da diese Information auch nach dem Löschen des Benutzers erhalten bleiben soll. Andernfalls würde sie dann verloren gehen.

Abbildung 47: Benutzer- und Qualitätsverwaltung



Das Ursprungsland (*sourceCountry*) wird nur bei der Qualität auf „kopiert“ gesetzt und ebenfalls nur als Zeichenkette hinterlegt. Gleichzeitig wird bei einer Kopie die Historie zurückgesetzt und fängt dann mit *COPIED* statt *NEW* an. Dies ist notwendig, um ein zu starkes Wachstum der Versionsgeschichte eines Landes zu verhindern. Ein Informationsverlust tritt aber nicht auf, da man in den seltenen Fällen, in denen man diese Informationen benötigt, diese im Ursprungsland findet. Erst wenn dieses Land gelöscht wurde, ist diese Information nicht mehr verfügbar. In diesem Fall dürften diese Informationen aber geringe Relevanz besitzen, sodass dies zugunsten eines geringeren Speicherplatzbedarfs in Kauf genommen wird.

*MODIFIED* wird benutzt, solange sich das Land in Bearbeitung befindet. Zur Protokollierung der genauen Änderungen gibt es die Möglichkeit, bei jeder Speicherung einen Kommentar hinzuzufügen. *VALIDATED* zeigt an, dass das Land fertig ist und überprüft wurde. Es steht normalerweise am Ende einer Versionshistorie. Nachträgliche Änderungen sind aber weiterhin möglich.

Die Rolle *READ\_ONLY* wird zurzeit nicht verwendet, da der lesende Zugriff momentan auch ohne explizites Login<sup>212</sup> erlaubt ist. In Zukunft könnte sich dies evtl.

<sup>212</sup> Für das Webinterface ist ein Login zurzeit auch für den lesenden Zugriff notwendig.

ändern und ist deswegen schon vorhanden. Alle anderen Rollen dürfen Daten verändern. Der *HELPER* hat jedoch kein Recht diese zu validieren. Der *Administrator* hat zudem das Recht, Benutzer anzulegen und SQL-Abfragen im Webinterface zu erstellen.

## 5.6 Finales Datenbankschema

### 5.6.1 Übersicht

Abbildung 48 auf Seite 122 zeigt eine Übersicht über das finale Datenbankschema. Abbildung 49 stellt die linke Seite des Schemas dar. Wird die oberste Zeile von links nach rechts gelesen, wird der grobe Aufbau ersichtlich. Ein Ländermodell enthält ein Unternehmensmodell, das auf den steuerspezifischen Teil verweist. Dieser beinhaltet wiederum eine Referenz auf die Steuerdaten, die auf die einzelnen Steuerobjekte verweisen. Die darunter liegenden Tabellen zeigen die zusätzlichen Beziehungen der obigen Elemente.

Abbildung 50 zeigt die rechte Seite des Schemas mit den möglichen Bemessungsgrundlagen der Steuerobjekte. Links oben ist *taxbasesammelbaustein*, als zentrale Komponente der Steuermodellierung zu sehen. Daneben sind alle anderen Bemessungsgrundlagen von oben nach unten abgebildet. In beiden Detailansichten wurden abbildungsübergreifende Beziehungen abgeschnitten. Diese sind in der Übersicht sichtbar.

Zentrales Element ist das *countrymodel*, das die Attribute eines Landes speichert. Die 1:n Beziehungen zu den Dateien für die Dokumentation (*pdffile*) und die Versionshistorie (*modificationhistoryelement*) sind in diesen Tabellen hinterlegt, indem sie auf den Schlüssel (*cm\_name*, *cm\_type*, *cm\_year*, *cm\_comment*) des Ländermodells verweisen.

Die einzige 1:1 Beziehung verweist auf das *unternehmensmodell*. Dieses enthält keine eigenen Daten, sondern verweist direkt weiter auf das *unternehmensmodellVst*. Im Prinzip könnte dieser Zwischenschritt entfallen und es könnte direkt auf das *unternehmensmodellvst* verwiesen werden. Dies ist jedoch der Programmstruktur geschuldet, da das Unternehmensmodell zur Trennung zwischen ökonomischen und steuerlichen Daten benötigt wird.



Neben der Formel für den Aufteilungsschlüssel, der gewählten Verbrauchsfolge und weiteren Wahlrechten, enthält es direkte Verknüpfungen zu den Daten der betrieblichen Altersvorsorge (*bavvstdaten*) und dem für das Unternehmen relevante Steuersystem (*steuerdatenunt*). Über Verknüpfungstabellen werden die Verbindungen (*unternehmensmodellvst\_bewertungbilanzobjekt* - Abschreibungen) zu den Bewertungsobjekten (*unternehmensmodellvst\_basicvalueobjekt* - Herstellungskosten) hergestellt. *Bewertungbilanzobjekt* enthält einen Verweis auf die gewählte Abschreibungsmethode (*universalmethode*). Des Weiteren ist jeder Anteilseigner einem *unternehmensmodellvst* zugeordnet. Die Tabelle *anteilseignervst* verweist ebenfalls auf ein Steuersystem aus der Liste der verfügbaren Systeme, die über die *umodell\_id* aus *steuerdaten* rekonstruierbar ist.<sup>213</sup>

Welches Steuerobjekt zu welchen Steuerdaten gehört, wird durch das Fremdschlüsselattribut spezifiziert. Durch *so\_pos* wird die Position in der Liste vermerkt. Somit kann eine weitere Verknüpfungstabelle eingespart werden. Neben den verschiedenen Perioden, in der die Steuer berechnet werden soll, den Daten für die Herstellungskosten (funktionale Aufteilung und Einbezug) und der Frage, ob sie liquiditätswirksam ist, wird auch auf das zugehörige Bewertungsobjekt (*hktax\_id*) verwiesen. Des Weiteren leitet *collectioncompononet\_id* auf den Sammelbaustein weiter.

Die einzelnen Bemessungsgrundlagen und ihre zugehörigen Rechenoperationen sind im *bmgbaustein* hinterlegt. Dieser enthält die Listenposition und den Fremdschlüssel des Sammelbausteins und erspart somit auch eine zusätzliche Verknüpfungstabelle. Über *tarifimp\_id* wird auf den Universaltarif verwiesen. Dessen verschiedene Stufen werden im *taxratelevel* verwaltet.

Durch den Polymorphismus kann die *taxbaseimp\_id* im *bmgb\_austein* auf alle sieben Bemessungsgrundlagen (*taxbasebalancesheet*, *taxbaseguv*, *taxbaseshareholder-income*, *taxbaseshareholderasset*, *taxbaseloscompensation*, *taxbaseanderesteuer*, *taxbasesammelbaustein*) verweisen. Die ersten vier verweisen über *tb\_basicdata* und den dazugehörigen Verknüpfungstabellen auf deren Elemente (*basicvalueobjekt*).

---

<sup>213</sup> Diese Liste ist Bestandteil des *UnternehmensmodellNst*. Die 1:n Beziehung wird auf der n-Seite abgebildet und erspart somit eine separate Verknüpfungstabelle.

Die Bemessungsgrundlage *taxbaseanderesteuer* leitet auf *othertaxeselement* weiter, das ein Steuerobjekt mit einer Rechenoperation und der Listenposition verknüpft.

Die Tabelle für das Verlustverrechnungsmodul verweist auf *losscompensationdata*. Neben den konkreten Daten zur Verwaltung der Verluste, enthält die Tabelle für die Vorschriften zur Ermittlung der verwendbaren Verluste und der Gewinne pro Periode auch zwei Verknüpfungen zum Sammelbaustein.

Der Tarif enthält auch eine Verbindung zu den PDF-Dokumenten und das Steuerobjekt zu den Elementen der Versionshistorie.

### 5.6.2 Besonderheiten

Die Tabellen *vitaxdatabaseuser*, *taxbaseshareholderincome* und *customview* in Abbildung 50 mögen sich abheben, da sie scheinbar keine Verbindung zu anderen Relationen haben. Wie bereits in 5.5 erwähnt, wird in der Historie ein direkter Verweis auf den Benutzer vermieden. Es wird nur dessen Name als Zeichenkette gespeichert. Daher existiert keine Fremdschlüsselbeziehung.

Bei der zweiten Tabelle konnte die Vererbungsstrategie „Single Table“ (vgl. 4.6) nicht verwendet werden, da in der Vererbungshierarchie bereits auf einer höheren Ebene (*TaxBaseImp* – „Table per Class“) eine andere gewählt wurde. Könnte die Vererbungsstrategie „Single Table“ benutzt werden, würde die Tabelle *taxbase-shareholderincome* gänzlich entfallen und alle Daten könnten von *taxbase-shareholderasset* gespeichert werden. Dies ist nicht möglich, sodass zwei identische Tabellen entstehen und sich die Daten auf beide Tabellen verteilen. Diese Daten werden jedoch nicht doppelt gespeichert, sodass sich die Redundanz auf die doppelte Tabellendefinition beschränkt. Durch den Polymorphismus<sup>214</sup> gelten alle Verbindungen von *taxbaseshareholderasset* entsprechend. Im Diagramm sind diese von der verwendeten Anwendung nicht explizit eingezeichnet, sodass diese Tabelle in der Darstellung des Schemas keine direkte Verbindung zu anderen Tabellen hat. Ebenso haben auch alle anderen Bemessungsgrundlagen keine ersichtlichen Verknüpfungen zum *bmg\_baustein*, da dieser auf die Oberklasse *TaxBaseImp* verweist.

---

<sup>214</sup> Die durch *taxbaseshareholderincome* abgebildeten Objekte sind eine Spezialisierung der Objekte aus *taxbaseshareholderasset* und erben somit auch alle Beziehungen der Oberklasse.

In der letzten Tabelle werden aus Vereinfachungsgründen die eingegebenen SQL-Abfragen (vgl. 6.9.2) für die Ansichten des Webinterface gespeichert. Diese haben mit dem eigentlichen Datenmodell nichts zu tun. Deswegen besitzt diese Relation auch keinerlei Beziehungen zu anderen Tabellen.

### 5.6.3 Formale Anforderungen

Die **erste Normalform** (NF) fordert, dass es keine sich wiederholenden Attribute oder Attributmengen<sup>215</sup> gibt. Bei strikter Auslegung könnte das Attribut *perioden* aus der Tabelle *steuerobjekt* dieser Anforderung widersprechen, da es sich aus neun<sup>216</sup> boolean-Variablen<sup>217</sup> zusammensetzt. Dieses wird jedoch als Einheit behandelt und in nur einer Spalte gespeichert, sodass bei der konkreten Umsetzung die erste Normalform erfüllt ist. Aus entwurfstheoretischer Sicht würde die optimale Lösung in einer separaten Tabelle bestehen. Für die sehr kleine Menge an Daten<sup>218</sup> wäre jedoch eine gesonderte Tabelle mit einer numerischen Spalte zur Rekonstruktion der Reihenfolge und eine zusätzliche Verknüpfungsoperation notwendig. Dies erscheint aus Gründen der Performanz und dem verhältnismäßig hohen zusätzlichen Bedarf an Speicherplatz nicht sinnvoll.

Die **zweite Normalform** fordert die Eliminierung von partiellen Abhängigkeiten von Attributen des Schlüssels. Durch die i. d. R. verwendeten synthetischen Schlüssel sind Teilabhängigkeiten ausgeschlossen, da die Schlüssel aus genau einem Attribut bestehen. Durch die Synthese gibt es auch keine echten funktionalen Abhängigkeiten. Somit sind nur noch die natürlichen Schlüssel und Join-Tabellen zu prüfen. Der einzige natürliche Schlüssel wird für das Land benutzt. Er besteht aus den Attributen

---

<sup>215</sup> „Es wird also verlangt, dass Attribute nicht weiter zerlegbar sind.“ Kemper, A./Eickler, A., 2009, S. 183. Ein Ehepaar kann z. B. mehr als ein Kind haben, daher sollte die Relation nicht für jedes Kind eine separate Spalte der Art „Kindname 1“, „Kindname 2“ usw. vorsehen. Dies wäre schon alleine dadurch problematisch, weil die erlaubte maximale Anzahl an Kindern ex ante nicht bekannt ist. Dies kann durch sogenanntes „Flachklopfen“ (Kemper, A./Eickler, A., 2009, S. 183.) behoben werden, indem die komplette Zeile bis auf den Kindernamen dupliziert wird. Sollten die Kinder mehr als ein Attribut besitzen (z. B. Name und Geburtsdatum), muss eine separate Relation angelegt werden und das Kindernamensattribut in der Tabelle der Eltern durch einen Fremdschlüssel ersetzt werden. Die Redundanz, die durch nahezu identische Zeilen entsteht, wird durch die nachfolgenden Normalformen vermieden bzw. reduziert.

<sup>216</sup> Die zehnte Periode wird über *lastperiod* abgebildet. Dies ist für mögliche Schlussbesteuerungen sinnvoll, falls sich der Simulationszeitraum ändern würde.

<sup>217</sup> Diese können nur die Werte wahr oder falsch annehmen und werden i. d. R. durch ein Bit (0 oder 1) abgebildet.

<sup>218</sup> Diese besteht aus 9 Bit.

*name*, *type*, *year* und *comment* (vgl. 5.3.2). Die restlichen Attribute *wechselkurs*, *displayname* und *unternehmensmodell\_id* sind direkt funktional abhängig von der Gesamtheit des Schlüssels

Intuitiv würde man zwar vermuten, dass sich bereits aus dem gewählten Land und dem jeweiligen Jahr der Wechselkurs<sup>219</sup> unmittelbar bestimmen lässt, und somit eine partielle funktionale Abhängigkeit vorliegt. Als Gegenbeispiel seien hier die Kaimaninseln genannt, die als britisches Überseegebiet eine eigene Währung, den Kaiman-Dollar, besitzen. In der vorliegenden Systematik würde man diese Inseln als Provinz von Großbritannien modellieren, sodass erst aus allen vier Attributen (→ keine teilweise Abhängigkeit) des Schlüssels sich der Wechselkurs bestimmen lässt. Sicherlich sind auch andere Auslegungen möglich, jedoch würde die explizite Abbildung dieser Abhängigkeit die Flexibilität der Datenstruktur nur unnötig beschränken. Die meisten Join-Tabellen enthalten nur die Fremdschlüssel der beteiligten Relationen. Bei einigen n:m-Beziehungen gibt es zusätzlich noch den *mapkey*, der das Schlüsselattribut der zur Abbildung der Beziehung verwendeten Map (assoziativen Array) speichert. Für geordnete Beziehungen gibt es ein numerisches Attribut. Diese sind dann jedoch Teil des Schlüssels. Somit gibt es auch hier keine teilweisen funktionalen Abhängigkeiten und jeder Schlüssel stellt bereits den minimal möglichen dar.

Die **dritte Normalform** fordert zusätzlich, dass es keine funktionalen Abhängigkeiten innerhalb der Menge von Nicht-Schlüsselattributen gibt. Beim ersten Entwurf wurde dies z. B. bei den *bavvstdaten* verletzt, denn aufgrund der jetzigen Programmstruktur impliziert z. B. die Nichtauswahl der beiden Durchführungsformen Pensionskasse und unmittelbare Zusage, dass Rückstellungen zu bilden sind.<sup>220</sup> Daher wurden die ursprünglichen drei Boolean-Variablen durch einen Aufzählungstypen (vgl. Abbildung 25) ersetzt. Somit ist die dritte Normalform und somit die Anforderungen aus Kapitel 3.1 erfüllt.

---

<sup>219</sup> Unterjährige Veränderungen werden nicht berücksichtigt. Zudem wird der Wechselkurs nur selten angepasst, um die Vergleichbarkeit zwischen den einzelnen Jahren zu gewährleisten. Somit existiert diese funktionale Abhängigkeit tatsächlich gar nicht.

<sup>220</sup> Im Benutzerdialog wird dies durch sogenannte *ButtonGroups* sichergestellt, innerhalb einer solchen Gruppe kann immer nur ein Element selektiert sein.

Bei der Erzeugung des Schemas stand jedoch der bereits existierende Programmcode im Vordergrund. Die Optimierung erfolgte in erster Linie durch Mapping-Optionen. Trotzdem wurde natürlich bei den vielfach notwendigen Refaktorisierungen und teilweisen Neuentwürfen auch auf die Implikationen für die Datenbank geachtet.

Das vorliegende Konzept stellt einen Kompromiss zwischen Performanz (optimal wäre dann eine einzige Tabelle) und den Anforderungen an ein gutes relationales Schema dar.

**Abbildung 48: Finales Datenbankschema -- Übersicht**

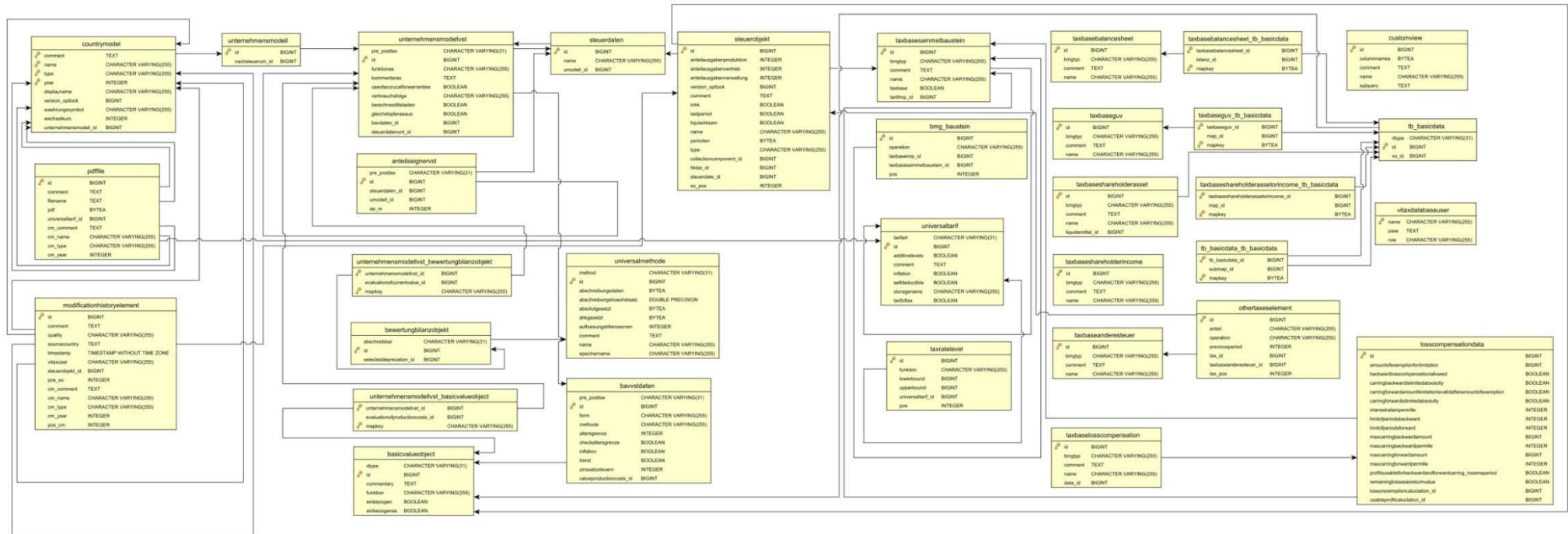
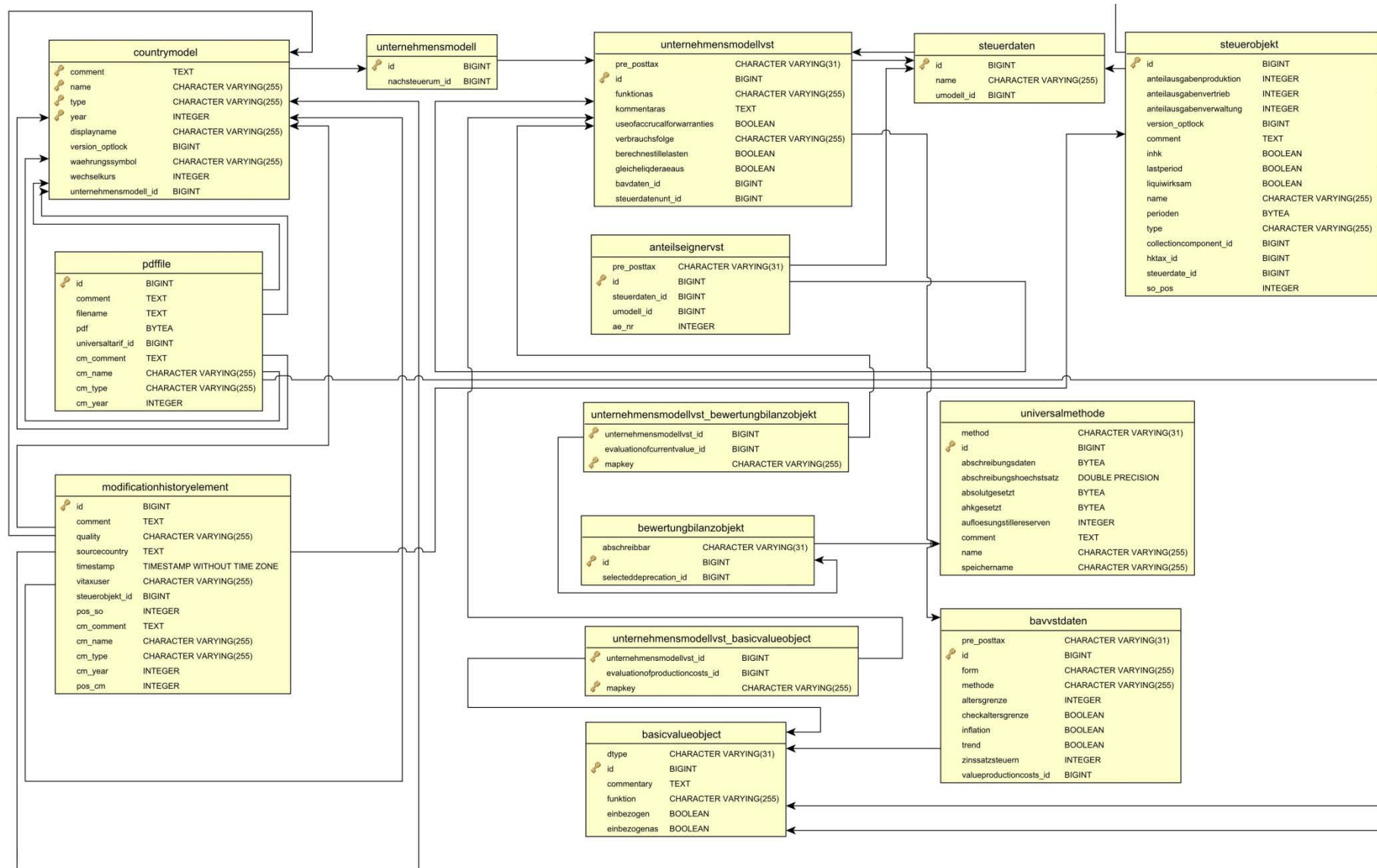
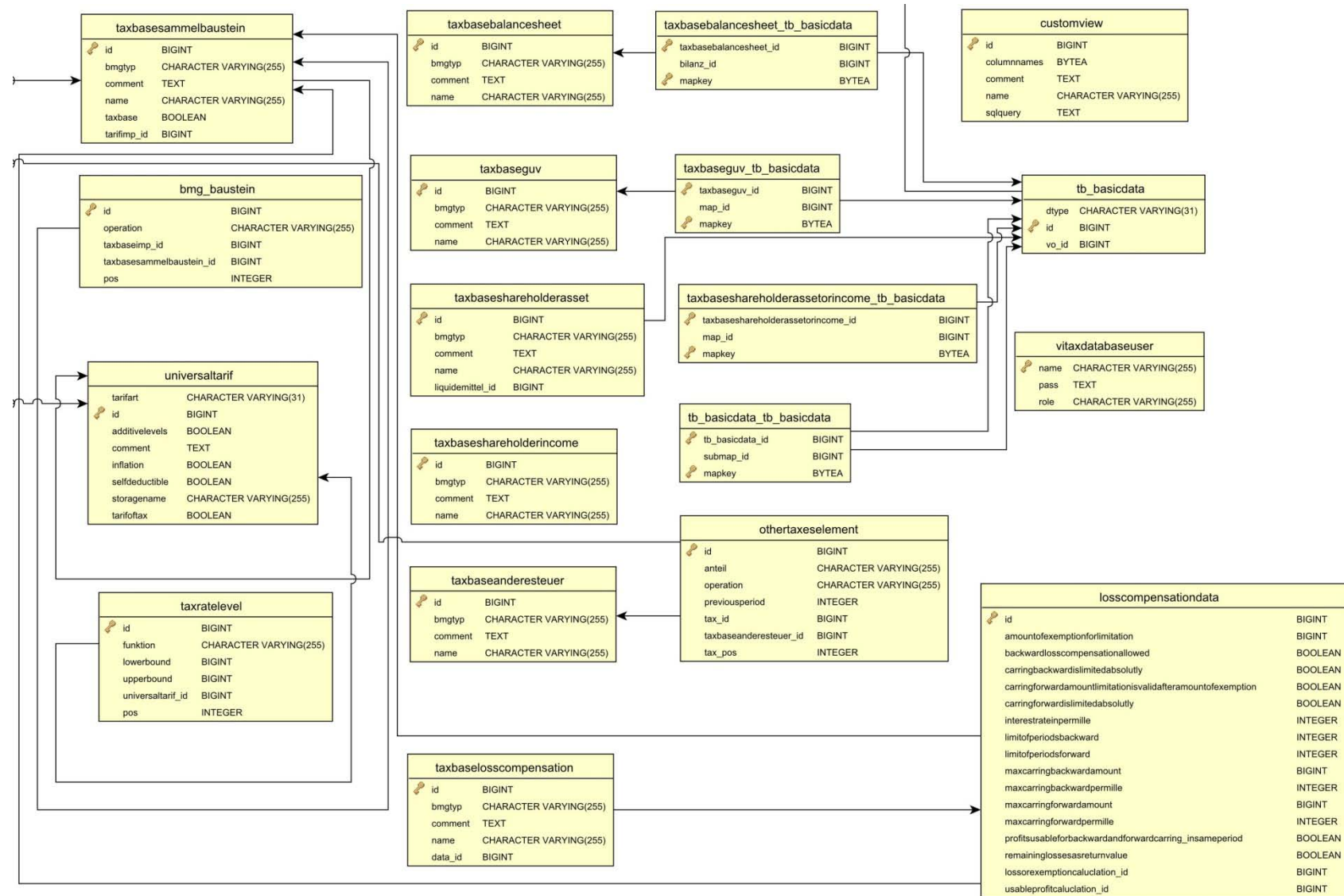


Abbildung 49: Teil des Datenbankschemas I/II -- Detailansicht



**Abbildung 50: Teil des Datenbankschemas II/II -- Detailansicht**



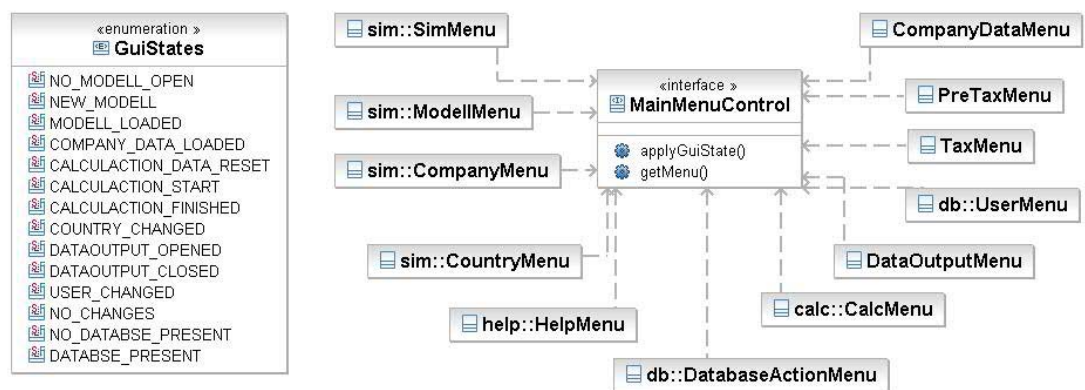


## 5.7 Sonstige Modifikationen

### 5.7.1 Ereignissystem für Menüelement

Zur Bewältigung der vielfältigen Kombinationen, wann welches Menüelement aktiv oder inaktiv sein soll, wurde das Menü in mehrere Klassen aufgespalten und ein Ereignissystem eingeführt. Solange z. B. keine Berechnung durchgeführt worden ist, können auch die Ergebnisse nicht aufgerufen werden. Daher darf dieses Menüelement erst nach dem Ende der Berechnungen verfügbar sein.

Abbildung 51: Ereignissystem der Menüstruktur und definierte Ereignisse



*GuiStates* (vgl. Abbildung 51) definiert die aktuell bekannten Ereignisse oder Zustände, die das System annehmen oder auslösen kann. Das Interface *MainMenuControl* definiert zur Unterstützung zwei Methoden. Dieses wird von allen Untermenüs (z. B. *SimMenu* oder *PreTaxMenu*) implementiert. Mithilfe von *getMenu* wird in der Klasse *MainFrame*, die als Singleton<sup>221</sup> implementiert wurde, die Menüleiste aufgebaut. Über diese Klasse können dann andere Programmteile einen Zustandswechsel propagieren. Durch den Aufruf von *applyGuiState* werden dann die Untermenüs dazu aufgefordert, ihren Zustand entsprechend zu ändern.

Ob die notwendigen Aktionen auch vom vorherigen Zustand abhängig sind, kann jede Umsetzung selbst entscheiden und den aktuellen Zustand entsprechend zwischenspeichern. Eine globale Speicherung findet nicht statt. Nach dem Schließen der Resultate sollten z. B. alle vorher wählbaren Elemente auch wieder verfügbar sein. Daher macht es einen Unterschied, ob es sich um ein neues Modell handelt oder ob

<sup>221</sup> Dieses Design-Pattern garantiert, dass es nur eine Instanz dieser Klasse geben kann.

dies geladen wurde, da nur im zweiten Fall der Menüpunkt *save* verfügbar wäre. Beim Start der Anwendung ist der Zustand *NO\_MODEL\_OPEN*. Abhängig davon, ob die Datenbankverbindung aufgebaut wurde oder nicht, wird das entsprechende Ereignis ausgelöst. Dieses entscheidet dann darüber, ob die datenbankspezifischen Menüelemente aktiv sind oder nicht.

Bei einem einfachen Anwendungsfall würde man z. B. ein Modell laden, dieses berechnen und sich danach die Resultate ansehen. Beim Laden würde das Ereignis *MODELL\_LOADED* von der Laderoutine erzeugt und propagiert. Dadurch würden die Menüelemente zur Eingabe und Modifikation von Parametern (*CompanyDataMenu*, *PreTaxMenu*, *TaxMenu*), die *Save-Methode*<sup>222</sup> und das Berechnungsmenu (*CalcMenu*) aktiviert. Nach dem Start der Berechnungen sind keine Eingaben mehr möglich. Erst nach Abschluss (vgl. *GuiStates* – *CALCULATION\_FINISHED*) sind dann die vorherigen Elemente wieder nutzbar und das Ausgabemenü (*DataOutputMenu*) ist verfügbar.

Somit kann nun detailliert auf einer geringen Stufe der Komplexität entschieden werden, wann einzelne Komponenten zu aktivieren oder zu deaktivieren sind. Dadurch sinkt die Wahrscheinlichkeit, Aktionen durch fälschlicherweise aktive Elemente auszulösen, die zurzeit nicht erlaubt oder nicht definiert sind. Zudem wird so im Programmcode dokumentiert, welche Zustände oder Ereignisse ausgelöst werden. Des Weiteren ist es nun nicht mehr notwendig, dass jedes Menüelement auf jedes andere direkten Zugriff haben muss. Somit ist eine stärkere Modularisierung möglich.

### 5.7.2 Zentralisierte Speicherung

Bisher war es oft der Fall, dass jedes Menüelement eine separate Speicher- bzw. Laderoutine implementiert hat.

*XML\_WriteOperations* und *XML\_ReadOperations* (vgl. Abbildung 52) sind abstrakte Klassen, die das *ActionListener*-Interface<sup>223</sup> implementieren, jedoch nicht die Methode *actionPerformed*. Dies hat den Hintergrund, dass die benötigten *Listener* diese

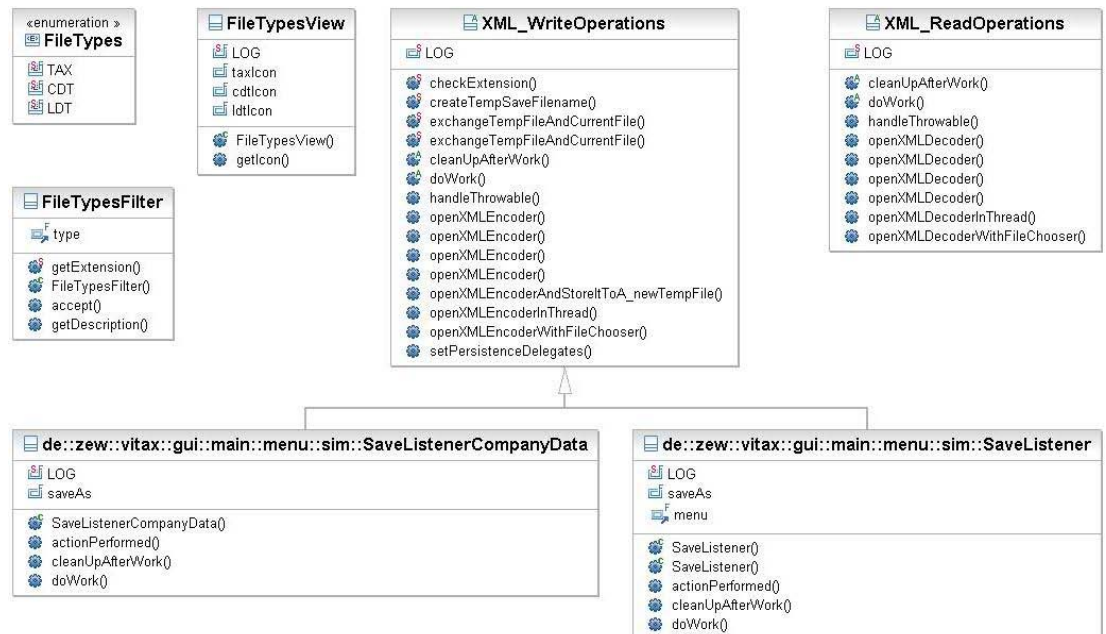
---

<sup>222</sup> Dadurch kann das Modell unter dem gleichen Namen gespeichert werden.

<sup>223</sup> Siehe `java.awt.event.ActionListener`.

Klassen erweitern und ihren spezifischen Code dort hinterlegen sollen. Ebenso stellen *doWork* und *cleanUpAfterWork* abstrakte Methoden dar, die entsprechend implementiert werden müssen. Diese werden von den *openXYZ*-Methoden aufgerufen.

Abbildung 52: Zentrale Infrastruktur zum Speichern und Laden von XML-Dateien



Der Ablauf stellt sich i. d. R. wie folgt dar: Zunächst trifft die konkrete Implementierung ihre Vorbereitungen (z. B. das Deaktivieren der Oberfläche oder die Initialisierung einer Fortschrittsanzeige) in der Methode *actionPerformed* und ruft von dort eine der *open*-Methoden auf. Diese rufen dann die *doWork*-Methode, in welcher der eigentliche Lade- oder Speichervorgang erfolgen soll, auf. Danach wechselt die Kontrolle wieder zurück zur *open*-Methode und ruft, nachdem sie die benötigten Ressourcen freigegeben hat, die *cleanUpAfterWork*-Methode auf. Hier können Vorbereitungen aus der *actionPerformed*-Methode wieder rückgängig gemacht werden und das entsprechende Ereignis für die Menüstruktur (vgl. 5.7.1) ausgelöst werden. Ebenso erfolgt die Fehlerbehandlung durch die *open*-Methoden.

Dateiauswahldialoge können durch die Wahl der entsprechenden *open*-Methode vorgeschaltet werden. Die Speicherung kann auch als eigener Thread (dt. leichtgewich-

tiger Prozess) ausgeführt werden, um das Aktualisieren von GUI-Elementen<sup>224</sup> zu ermöglichen.

Durch diese Maßnahmen ist es möglich, den Umfang der konkreten Implementierungen auf das Wesentliche zu reduzieren und so die Lesbarkeit deutlich zu erhöhen. Zudem können immer wieder benötigte Aktionen und Konfigurationen in der abstrakten Klasse realisiert werden und müssen nicht mehr dupliziert werden. Dadurch muss die Korrektheit dieser Programmteile nur noch einmal validiert werden.

Die beiden *SaveListener* sind nur zwei ausgewählte Realisierungen. Sie werden mehr als einmal benutzt. Alle anderen Umsetzungen erfolgen in anonymen Klassen und sind deshalb nicht Bestandteil des Klassendiagramms.

Die *FileType*-Klassen werden für die Dateiauswahldialoge benutzt, um festzulegen, welche Dateitypen (*FileTypes*) angezeigt (*FileTypesFilter*) und mit welchen Icons (*FileTypeView*) sie verknüpft werden sollen.

### 5.7.3 Tabellen für Ausgabemasken

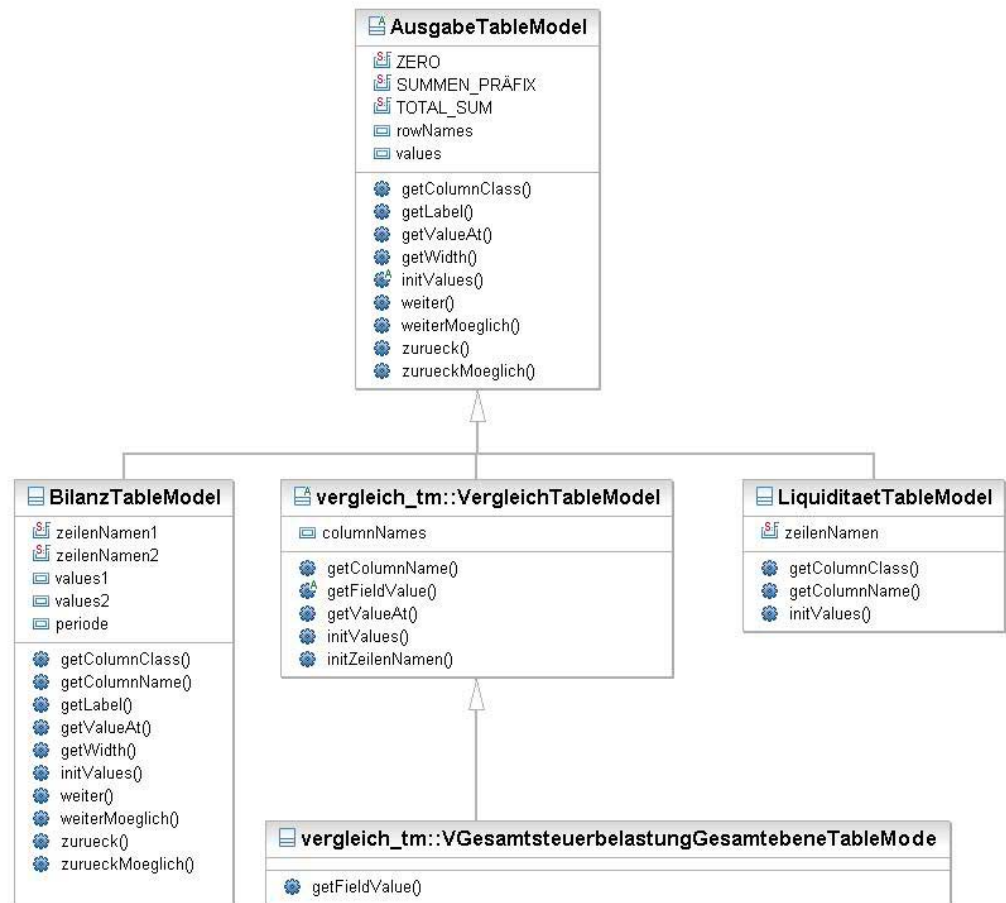
Zur Aufbereitung der komplexen Berechnungsergebnisse werden sehr viele unterschiedliche Tabellen zur Anzeige benötigt. Um den Implementierungsaufwand möglichst gering zu halten, wurden verschiedene abstrakte Klassen implementiert, so dass i. d. R. nur eine Methode zur Initialisierung der Tabelleninhalte überschrieben werden muss.

Abbildung 53 enthält nur einige wenige konkrete Implementierungen. Das *LiquiditaetTableModel* repräsentiert den Standardfall, in dem nur eine Tabelle angezeigt wird. Dann genügt es in den meisten Fällen, die *initValues*-Methode zu implementieren. Das *BilanzTableModel* repräsentiert einen etwas komplizierteren Spezialfall, bei dem mehrere Tabellen durchgeblättert werden können; hier z. B. die Bilanzen der einzelnen Perioden. Dafür müssen zusätzlich die Navigationsmethoden (z. B. *weiter* und *zurück*) implementiert werden.

---

<sup>224</sup> Insbesondere bei Verlaufsanzeigen besteht das Problem, dass diese nicht aktualisiert werden, solange die Event-Queue nicht verlassen wurde.

Abbildung 53: Tablemodels für die Datenausgabe



Für die Vergleichstabellen wird bereits eine standardisierte *initValues*-Methode bereit gestellt, die durch alle berechneten Länder iteriert und entsprechend *getFieldValue* aufruft. Somit muss nur noch *getFieldValue* überschrieben werden, um die konkreten Werte für die übergebenen Argumente, z.B. die Gesamtsteuerbelastung für das aktuelle Land, zu erhalten.

## 6 Fallbeispiel

### 6.1 Sachverhalt

Anhand eines konkreten Beispiels wird abschließend die komplette Anwendung inkl. des Webinterfaces dargestellt und deren Funktionsweise demonstriert. Schritt für Schritt wird sukzessive ein ökonomisches Modell aufgebaut. Danach werden die steuerlichen Informationen anhand der Daten aus einer Studie des ZEWs für die Industrie und Handelskammer (IHK) Stuttgart<sup>225</sup> für Deutschland hinzugefügt. Diese untersuchte „die Entwicklung der direkten und indirekten Substanzbesteuerung“<sup>226</sup> über einen Zeitraum von 1990 bis 2009 in qualitativer und quantitativer Hinsicht. Der Begriff Substanzbesteuerung umfasst hierbei alle Steuern, die nicht gewinnabhängige Bestandteile besitzen. Neben den direkten Substanzsteuern wie z. B. der Vermögensteuer, der Grundsteuer und der Gewerbesteuer vom Kapital, betrifft dies auch die Gewerbesteuer vom Ertrag und die Körperschaftsteuer infolge der Hinzurechnung von bestimmten Aufwandspositionen zu den jeweiligen Bemessungsgrundlagen. Bei der Gewerbesteuer werden z. B. Teile der Dauerschuldzinsen<sup>227</sup> der Bemessungsgrundlage hinzugerechnet. Für die quantitative Beurteilung wurden die Jahre 1990, 1993, 1998, 2004 und 2009 mit dem European Tax Analyzer berechnet, da zu diesen Zeitpunkten signifikante Änderungen bei den Regelungen zur Substanzbesteuerung in Kraft traten.<sup>228</sup> Diese partielle Zeitreihe soll nun erweitert werden, so dass im Ergebnis alle Daten für die Berechnung einer Zeitreihe von Deutschland von 1990-2009 in der Datenbank vorliegen.

Im Folgenden wird die Möglichkeit zur Recherche der wesentlichen Steuersätze mithilfe des Webinterfaces demonstriert. Abschließend werden noch verschiedene Szenarien analysiert. Zum einem wird die Simulation einer real- und finanzwirtschaftli-

---

<sup>225</sup> Spengel, C., et al., Substanzbesteuerung, 2010.

<sup>226</sup> Spengel, C., et al., Substanzbesteuerung, 2010, S. 15.

<sup>227</sup> „Hervorzuheben ist hier die Neugestaltung der Hinzurechnungstatbestände in § 8 Nr. 1 GewStG, die insbesondere den zuvor geltenden § 8 Nr. 1 GewStG [1990-2007] (50%ige Hinzurechnung von Dauerschulden) sowie die Vorschriften des § 8 Nr. 2, 3 und 7 GewStG [1990-2007] durch die 25%ige Hinzurechnung aller Zinsen und Finanzierungsanteile von Mieten, Pachten, Leasingraten und Lizenzen mit einem Freibetrag von EUR 100.000 (§ 8 Nr. 1 GewStG) ersetzt hat.“ Spengel, C., et al., Substanzbesteuerung, 2010, S. 43.

<sup>228</sup> Vgl. Spengel, C., et al., Substanzbesteuerung, 2010.

cher Krisensituation durchgeführt und zum anderem werden zwei große Unternehmen aus verschiedenen Branchen betrachtet.

## 6.2 Benutzer anlegen

Zurzeit ist es lediglich erforderlich, sich als Benutzer zu autorisieren, wenn die Datenbasis der Datenbank verändert werden soll. Der Administrator, ein spezieller Benutzeraccount, vergibt solche Kennungen. Dieser Account wurde entweder bereits angelegt oder kann mit einer speziellen Version bzw. via SQL-Skript angelegt werden. Alle Nutzerdaten sind in der Tabelle *vitaxdatabaseuser* hinterlegt.

Abbildung 54: Anlegen eines Benutzers

In Abbildung 54 wird ein Benutzer namens „testUser“ angelegt. Der zugehörige Datenbankeintrag ist in Abbildung 55 Zeile 8 ersichtlich. Die Dialoge zum Anlegen und Löschen von Benutzern sind der Rolle *Administrator* vorbehalten. Das zu Beginn vergebene Passwort kann vom Benutzer nach seiner Anmeldung (vgl. Abbildung 56) jederzeit verändert werden (vgl. Abbildung 57). Diese Daten sind sowohl für die Desktop- als auch die Webanwendung gültig.

Abbildung 55: Beispieltabelle für Benutzerdaten

	name	pass	role
1	mgw	098f6bcd4621d373cade4e832627b4f6	ADMINISTRATOR
2	cer	098f6bcd4621d373cade4e832627b4f6	WORKER
3	kfi	098f6bcd4621d373cade4e832627b4f6	WORKER
4	bzi	098f6bcd4621d373cade4e832627b4f6	WORKER
5	f-gehr	098f6bcd4621d373cade4e832627b4f6	HELPER
6	lev	098f6bcd4621d373cade4e832627b4f6	WORKER
7	ddr	098f6bcd4621d373cade4e832627b4f6	WORKER
8	testUser	ad0234829205b9033196ba818f7a872b	HELPER

Quelle: Eigene Darstellung – Programm DbVisualizer

Wie aus Abbildung 55 ersichtlich ist, wird das Passwort verschlüsselt abgelegt, so dass selbst bei einem direkten Zugriff auf die Datenbank es nicht ohne weiteres ersichtlich oder veränderbar ist. Eine potentielle Schwachstelle besteht jedoch darin, dass sich bei direktem Zugriff ein bereits registrierter Benutzer eine andere Rolle zuweisen und anschließend mit erhöhten Rechten arbeiten könnte. Eine Verschlüsselung der Rollenbezeichnung würde in diesem Fall keine Abhilfe schaffen, da ein potentieller Angreifer lediglich alle möglichen Kombinationen (zurzeit vier) mit seinem Account testen müsste. Da dies ohne Probleme und mit einem geringen zusätzlichen Zeitaufwand möglich wäre, erscheint eine Verschlüsselung der Rollenbezeichnung nicht sinnvoll.

**Abbildung 56: Benutzeranmeldung**

Das vorliegende Benutzersystem soll jedoch primär den Zweck erfüllen, Änderungen an den hinterlegten Daten zu protokollieren und einzelnen Benutzern zuzuordnen, und nicht eine sichere Authentifikation an sich gewährleisten. Dieser Nachteil wird somit zurzeit zugunsten einer einfachen Lösung in Kauf genommen. Zudem ist der direkte Datenbankzugriff nur mithilfe der Kenntnis der notwendigen Verbindungsdaten inklusive eines zusätzlichen Passworts möglich. Aufgrund dessen ist dies sehr unwahrscheinlich und stellt kein isoliertes Problem dieser Lösung dar, sondern ist immer ein sicherheitstechnisches Problem.

**Abbildung 57: Dialog zum Wechseln des Passwortes**



### 6.3 Ökonomische Daten

Zunächst sind die Daten bzgl. der Vermögensstruktur und der Einkünfte der bis zu zehn Anteilseigner (vgl. Abbildung 58) anzugeben. Neben der Beteiligung an der Gesellschaft und der Quote am Gesellschafterdarlehn, sind auch andere Einkunftsarten (Geschäftsführergehalt, sonstige Arbeitseinkünfte und übrige Einkünfte), Vermögen (sonstiges Vermögen mit und ohne Inflationsausgleich, Barvermögen) sowie ein Konsumbetrag anzugeben. Je nach Untersuchungsziel müssen aber nicht alle Daten spezifiziert werden.

Abbildung 58: Eingabe der Anteilseignerdaten

Anteilseig...	Anteil am ...	Übrige Ei...	Sonstiges...	Konsumb...	GGF-Gehalt	Virtuelle E...	sonstige ...	Anfangsb...	Sonstiges...	Proportio...	Sharehol...	Number o...
1	51	0	25.000	0	0	25.000	0	0	0	0		0
2	9	0	25.000	0	0	25.000	0	0	0	0		0
3	5	0	25.000	0	0	25.000	0	0	0	0		0
4	5	0	0	0	0	25.000	0	0	0	0		0
5	5	0	0	0	0	25.000	0	0	0	0		0
6	5	0	0	0	0	25.000	0	0	0	0		0
7	5	0	0	0	0	25.000	0	0	0	0		0
8	5	0	0	0	0	25.000	0	0	0	0		0
9	5	0	0	0	0	25.000	0	0	0	0		0
10	5	0	0	0	0	25.000	0	0	0	0		0

Eine Besonderheit stellt das virtuelle Einkommen dar, da es keinerlei direkte Auswirkungen auf das Einkommen oder den Besitz hat. Es dient primär dazu, das zu versteuernde Einkommen ohne unerwünschte Nebeneffekte, wie z. B. die Erzeugung von Zinsen, zu erhöhen. Dies ist notwendig, um auf Anteilseignerebene (natürliche Personen) bei nicht proportionalen Tarifverläufen die gewünschte Tarifbelastung zu simulieren.

Danach werden die einzelnen Wirtschaftsgüter (wie z. B. Patente, Lizenzen Maschinen, Gebäude, Grundstücke und Beteiligungen) sowie die Finanzstruktur (wie z. B. Forderungen, Kassenbestand, Verbindlichkeiten, Rückstellungen und Eigenkapital des Unternehmens) aggregiert in die Bilanz (vgl. Abbildung 59) eingetragen.

Anschließend werden diese Größen in Detail-Dialogen weiter spezifiziert. Die Position der Grundstücke und Gebäude (vgl. Abbildung 60) untergliedert sich z. B. in ein Grundstück, ein Bürogebäude und ein Fabrikgebäude. Die Anschaffungskosten werden als prozentualer Teil der Gesamtposition angegeben. Zusätzlich ist eine wirtschaftliche Nutzungsdauer einzutragen. Diese bestimmt i. d. R. den Abschreibungszeitraum, falls es steuerlich keine anderen Vorgaben gibt, sowie den Zeitpunkt für eine mögliche Ersatzbeschaffung, falls keine vorherige Veräußerung geplant ist. Bei

bebauten Grundstücken ist noch der Anteil an Grund und Boden zu bestimmen, da dieser normalerweise nicht abgeschrieben wird.

**Abbildung 59: Eingabe der Bilanzdaten**

Assets		Liabilities	
<b>fixed assets</b>		<b>EK, FK</b>	
real estate	1030000	equity capital	350000
Machinery	2500000	accrual for pensions	0
production equipment	80000	Other provisions	0
office equipment	80000	Other long-term Liabilities	550000
Intangible Assets	20000	long-term liabilities against shareholders	720000
domestic debt	20000	Short-term liabilities	216
Foreign Investment	10000		
domestic Investment	30000		
Foreign debt	10000		
<b>working assets</b>			
Manufactured goods	0		
short-term debt	0		
Liquidity	0		

The balance-sheet is not balanced. The assets are by 2160000 too high.

OK Cancel

Zum Schluss müssen auch noch Informationen bzgl. der firmeninternen Nutzung des Wirtschaftsgutes hinterlegt werden, da diese Informationen zur Ermittlung der Herstellungskosten benötigt werden. Dafür muss ein prozentualer Anteil den funktionalen Bereichen Produktion, Verwaltung und Vertrieb zugeordnet werden.

**Abbildung 60: Eingabe der Details der einzelnen Bilanzpositionen**

**Weitere Angaben zu den einzelnen Bilanzposten**

Langfristige Forderungen (Inland) Langfristige Forderungen (Ausland) Inlandsbeteiligungen  
Pensionsverpflichtung Finanzierung durch Anteilseigner kurzfristige Verbindlichkeiten  
Geschäftsausstattung Immaterielle Wirtschaftsgüter Auslandsbeteiligungen  
**Grundstücke** Maschinen Betriebsausstattung sonstige langfristige Verbindlichkeiten

**Bisherige Eingaben**

Gesamtwert (Bilanz) 1030000  
Anzahl der Gegenstände 3

Unbebautes Grundstück Büro- und Geschäftsgebäude **Fabrikationsgebäude**

**Allgemeine Angaben**

Anteil am Gesamtwert [Prozent] 30  
Wirtschaftliche Nutzungsdauer [Perioden] 50  
Bisherige Nutzungsdauer [Perioden] 0

**Anteil der Nutzung an**

- Produktion 40 %  
- Verwaltung 40 %  
- Vertrieb 20 %

**Veräußerung**

Veräußerung durchführen ☐  
Eventuelle Veräußerungsperiode 0

**Zusätzliche Angaben**

Anteil von Grund und Boden [Prozent] 20

OK Cancel

Zur Fortschreibung dieser Daten werden noch Informationen über Preissteigerungsraten und Zinssätze (vgl. Abbildung 61) benötigt. Die Wiederbeschaffungskosten einer Maschine berechnen sich z. B. aus den inflationierten Anschaffungskosten. Ebenso erhöhen sich auch andere Kosten wie z. B. Löhne und Gehälter und Materialkosten. Bei Kostenarten, die kontinuierlich auftreten, wird in der aktuellen Periode jedoch nicht die volle Rate berechnet, sondern nur deren Wurzel, um eine sukzessive Preissteigerung abzubilden. Wurden z. B. Teile des benötigten Materials bereits zu Beginn des Zeitabschnitts gekauft, konnte es noch zu den alten Preisen erworben werden. Die Steigerungsrate der Lebenshaltungskosten wird für die Berechnung von Absatzpreisen, Rentenansprüchen und Einkünften der Anteilseigner benötigt.<sup>229</sup>

*Abbildung 61: Eingabe der Inflationsraten und Zinsen*

Eingabe in Promille	
primary products	14
cost of living	23
wage settlements	25
investment goods	25
industrial sites	25

Des Weiteren werden noch detaillierte Informationen über den Produktions- und Absatzplan (vgl. Abbildung 62), die Personalstruktur und -vergütung und über weitere Einnahmen bzw. Ausgaben des Unternehmens benötigt.

Aus all diesen Daten ergibt sich das Rahmenmodell des Unternehmens. Für einen aussagekräftigen Vergleich werden jedoch noch weitere Angaben benötigt. Diese umfassen die Abschreibungsregeln, die Herstellungskostenermittlung, die Verbrauchsfolge, die Durchführungsparameter der betrieblichen Altersvorsorge sowie die Wahl der Bildung einer Rückstellung für Gewährleistungsverpflichtungen. Diese

<sup>229</sup> Für detaillierte Informationen bezgl. der Fortschreibung vgl. Meyer, R., Simulation von Steuerbelastungen, 1996, S. 137 und 139.

zusätzlichen Spezifikationen werden auch für die Steuermodellierung benötigt und bei dieser auch näher erläutert.

*Abbildung 62: Eingabe des Unternehmensplans*

Dieses Rahmenmodell kann getrennt (\*.cdt) oder inklusive der Ländermodellierungen (Steuersysteme) als komplettes Modell (\*.tax) als XML-Datei gespeichert werden. Auch ein datenbankabhängiges Modell ist möglich, bei dem in der lokalen Datei nur die Schlüsselattribute des Landes hinterlegt werden und die Steuerdaten aus der Datenbank nachgeladen werden. Durch den Wechsel des Rahmenmodells (Laden einer anderen cdt-Datei) können dieselben Steuersysteme auf unterschiedliche ökonomische Ausgangsfälle angewendet werden.

#### 6.4 Erstellen des Steuersystems für 1990

In der oben angesprochenen Studie wurden die Jahre 1990, 1993, 1998, 2004 und 2009 für Deutschland untersucht. Als Erstes wird nun das steuerliche Modell für Deutschland 1990 (vgl. Abbildung 63) für den damals gültigen Rechtsstand erzeugt. Ein Kommentar, der die gewählte Variation näher beschreibt, wird in diesem Fall nicht benötigt und kann auch nicht eingegeben werden, um die Einzigartigkeit dieser Kombination zu gewährleisten. Der Wechselkurs beträgt 1,956 DM/€ Dieser wird benötigt, um Freibeträge oder Stufengrenzen, wie z. B. bei der Einkommensteuer, ohne manuelle Transformation abbilden zu können, da das Simulationsprogramm in Euro rechnet.

Abbildung 63: Erzeugen eines neuen Ländermodell

Abbildung 64 zeigt das Hauptmenü des European Tax Analyzer. In Deutschland ist es zulässig, für zukünftige Gewährleistungen eine Rückstellung<sup>230</sup> zu bilden. Für den Endvermögensvergleich sollen stille Lasten und Reserven (vgl. 2.2.2) am Ende des Planungshorizontes aufgelöst werden, um so Verzerrungen aufgrund von Bewertungsunterschieden (z. B. unterschiedliche Restbuchwerte bei Maschinen) zu vermeiden.

Abbildung 64: Auswahl der Gewährleistungsrückstellung



Danach sind die Durchführungsoptionen<sup>231</sup> für die betriebliche Altersvorsorge zu bestimmen. Im konkreten Fall dürfen Rückstellungen erst gebildet werden, wenn der begünstigte Mitarbeiter das dreißigste Lebensjahr<sup>232</sup> bereits vollendet hat. Zudem

<sup>230</sup> Vgl. § 6 Abs. 1 Nr. 3 a und b EStG.

<sup>231</sup> Vgl. § 6a EStG.

<sup>232</sup> Vgl. § 6a Abs. 1, 2 Nr. 1 EStG.

beträgt der Kalkulationszinssatz 6 %.<sup>233</sup> Entsprechend dem Stichtagsprinzip<sup>234</sup> dürfen nur bereits erworbene, d. h. unverfallbare Ansprüche berücksichtigt werden. Ebenso dürfen keinerlei Gehaltssteigerungen oder inflationsbedingte Rentenanpassungen antizipiert werden.

Das Gegenwartsverfahren und das Teilwertverfahren unterscheiden sich primär darin, auf welche Weise sie Änderungen des Barwertes der Versorgungsleistungen zum Zeitpunkt der Pensionierung berücksichtigen. Beim ersten Verfahren wird diese Änderung gleichmäßig auf die restlichen Jahre verteilt, während beim zweiten der auf die bisherigen Jahre entfallende Anteil in Form einer Einmalzuführung sofort nachgeholt wird. In Deutschland gilt das Teilwertverfahren.

**Abbildung 65: Eingabe der Durchführungsdetails für die Betriebliche Altersvorsorge**

In einem nächsten Schritt sind die Abschreibungen zu spezifizieren. Entsprechend der Modellparameter sind die zugehörigen steuerlichen Abschreibungszeiträume den AFA-Tabellen<sup>235</sup> zu entnehmen. Wie bereits in 5.3.5 erwähnt, gibt es mehrere Spezialisierungen der Universalmethode zur Auswahl, die die Eingabe deutlich vereinfachen. Dadurch entfällt die Notwendigkeit, die konkreten Abschreibungsraten manuell zu berechnen. Zudem wird so besser dokumentiert, welche Abschreibungsarten kodifiziert sind.

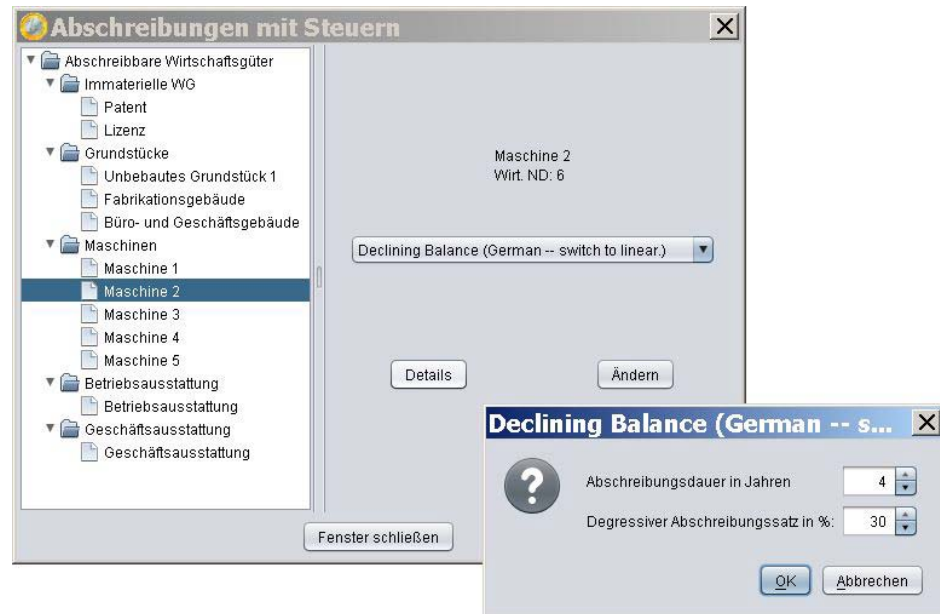
<sup>233</sup> Vgl. § 6a Abs. 3 Nr. 1 und 2 EStG.

<sup>234</sup> Vgl. § 6a Abs. 3 Nr. 1 Satz 2 EStG; Abschn. 41 Abs. 19 EStR.

<sup>235</sup> In der Abschreibungstabelle (kurz: "AFA-Tabelle") ist die betriebsgewöhnliche Nutzungsdauer (§ 7 Abs. 1 Satz 2 Einkommensteuergesetz) ausgewiesen. Vgl. Bundesministerium der Finanzen, <http://www.bundesfinanzministerium.de/.../AFA-Tabellen/005.html>, 18.07.2010.



Abbildung 66: Eingabe der Abschreibungen



Zur vereinfachten Bewertung des Lagerbestandes ist ein Verbrauchsfolgeverfahren festzulegen. Im Gegensatz<sup>236</sup> zur Vermögensteuer<sup>237</sup> ist Lifo (Last in first out) bei der Körperschaftsteuer bereits zulässig, d. h. es wird unterstellt, dass zuerst die Produkte verkauft werden, die als letztes eingelagert<sup>238</sup> wurden.

Abbildung 67: Eingabe des Verbrauchsfolgeverfahrens



Für diese Bewertung sind die Regeln zur Ermittlung der Herstellungskosten relevant. Abbildung 68 zeigt den zugehörigen Dialog. Hier kann bestimmt werden, welche Bestandteile in welcher Höhe zu berücksichtigen sind. Mit Hilfe von Formeln kann auf die funktionale Aufteilung der Kosten zurückgegriffen werden.

<sup>236</sup> § 6 Abs. 1 Nr. 2a EStG.

<sup>237</sup> § 11 Abs. 2 BewG und § 109 Abs. 4 BewG.

<sup>238</sup> Die Produktion der aktuellen Periode wird zunächst vollständig eingelagert. Aus dem Lager wird dann, entsprechend der Verbrauchsfolge, die notwendige Absatzmenge entnommen und bewertet.

Abbildung 68: Einbezug der Grundsteuer in die Herstellungskosten

**Herstellungskosten (Tax)**

**mögliche Bestandteile**

- Kurzfristige Verbindlichkeiten
- Langfristige Verbindlichkeit gegenüber Anteilseigner
- Sonstige langfristige Verbindlichkeiten
- Verbindlichkeiten aus LuL

**einbezogene Bestandteile**

- Betriebliche Altersvorsorge
- Betriebsausstattung
- Büro- und Geschäftsgebäude
- Energieverbrauch
- Fabrikationsgebäude
- FuE
- GdG Gehalt
- Geschäftsausstattung
- Grundsteuer**
- Lizenz
- Maschine 1
- Maschine 2
- Maschine 3
- Maschine 4
- Maschine 5
- Material
- Patent
- Personalraten
- Sonstige Ausgaben
- Unbebautes Grundstück 1

**Grundsteuer**

AntProdAbs  Validate formula

Anteil Produktion in % : AntProd  
Anteil Verwaltung in % : AntVerw  
Anteil Vertrieb in % : AntVert

Anteil Produktion in € : AntProdAbs  
Anteil Verwaltung in € : AntVerwAbs  
Anteil Vertrieb in € : AntVertAbs

Aufteilungsschlüssel: As

Gesamtaufwand in €: GesAuf

**Kommentar**

Vollkostenansatz Vorsteuer. Es fließen die kompletten Produktions- und Verwaltungskosten mit in die Herstellungskostenermittlung ein.

Close

Auch der Einbezug von Steuern ist möglich. Diese müssen jedoch explizit (vgl. Abbildung 72) dafür vorgesehen werden. Zudem sollten diese Steuern keine gewinnabhängigen Bestandteile beinhalten, da die Herstellungskosten direkten Einfluss auf den Gewinn haben und somit eine zirkuläre Abhängigkeit entsteht, die nicht auflösbar ist.

Abbildung 69: Eingabe der Steuern

**Edit tax system**

**Tax Data Company**

- Grundsteuer
- Einheitswert des Betriebsvermögens
- Vermögenssteuer
- Gewerbesteuer vom Kapital
- BMG Gewerbesteuer vom Ertrag
- Verlustmodul GewSt
- Gewerbesteuer vom Ertrag
- BMG Körperschaftsteuer
- Verlust Modul KSt Vorbereitung
- Verlust Rücktrag KSt
- Verlustvortrag
- Körperschaftsteuer
- Herstellung der Ausschüttungsbelastung
- Bestand\_EK\_01
- Bestand\_EK\_02
- Bestand\_EK\_50
- Auflösung Verlustmodul Gewerbesteuer in t=10
- Auflösung Verlustvortrag KSt in Periode 10

Name	Type	PC	liquidity	All Peri...	Last Mo...	Quality	PDF
Grundsteuer	REAL_PROPERTY_TAX	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Einheitswert des Betriebsvermögens	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Vermögenssteuer	OTHER_TAX_ON_CAPITAL	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Gewerbesteuer vom Kapital	TRADE_TAX_ON_CAPITAL	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
BMG Gewerbesteuer vom Ertrag	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Verlustmodul GewSt	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Gewerbesteuer vom Ertrag	TRADE_TAX_ON_INCOME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
BMG Körperschaftsteuer	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Verlust Modul KSt Vorbereitung	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Verlust Rücktrag KSt	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Verlustvortrag	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Körperschaftsteuer	CORPORATE_INCOME_TAX	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Herstellung der Ausschüttungsbelastung	SURCHARGE_TAX	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Bestand_EK_01	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Bestand_EK_02	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Bestand_EK_50	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Auflösung Verlustmodul Gewerbesteuer in t=10	AUXILIARY_CALCULATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>
Auflösung Verlustvortrag KSt in Periode 10	OTHER_TAX	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mgw	MODIF...	<input type="checkbox"/>

**Tax controls**

Add Tax  
Edit Tax  
Delete Tax  
Mark as ...

**Error Messages!**

**Comment for selected tax.**

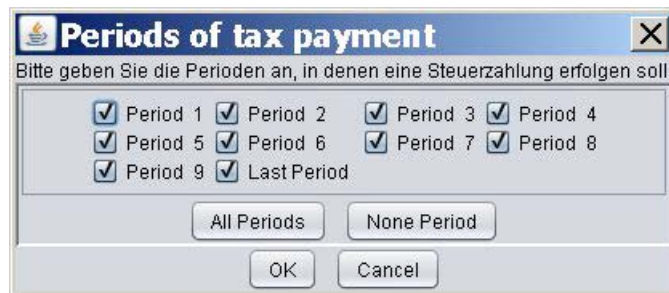
Close

Der mit Abstand aufwendigste Teil der Modellierung eines Landes sind die Steuern selbst, da zur adäquaten Berechnung der Bemessungsgrundlagen häufig komplexe Nebenrechnungen notwendig sind. Diese können aus Gründen der Übersichtlichkeit und der Wiederverwendung (vgl. 5.3.7.2.2) auch als eigene Steuern modelliert wer-



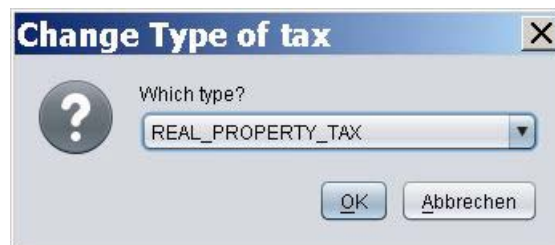
den, welche dann jedoch als nicht liquiditätswirksam gekennzeichnet werden müssen, um nicht die Steuerzahlung zu beeinflussen. Des Weiteren ist es möglich, die Berechnung für einzelne Perioden (vgl. Abbildung 70) auszusetzen oder nur in diesen Perioden zu berechnen. Im Allgemeinen wird diese Möglichkeit für die Steuerartenrechnung benutzt, bei der sukzessive eine Steuerart nach der anderen ausgeschaltet wird, um so den Anteil einer Steuer an der Gesamtbelastung zu bestimmen.<sup>239</sup>

*Abbildung 70: Auswahl der Berechnungsperioden einer Steuer*



Für die vereinfachte Recherche und die automatisierte Steuerartenrechnung ist es notwendig, die Steuerart zu spezifizieren. Zudem ist es auf diese Weise auch einfacher Hilfsrechnungen zu identifizieren.

*Abbildung 71: Auswahl der Steuerart*



Wie bereits oben erwähnt, ist es möglich, eine Steuer als Bestandteil (vgl. Abbildung 72) der Herstellungskosten zu deklarieren. Dafür muss auch eine Aufteilung der Steuerzahlungen auf einzelne Funktionsbereiche vorgenommen werden. Diese Werte sind dann beim Einbezug in die Herstellungskosten (vgl. Abbildung 68) als Variable verfügbar.

Außerdem ist in der Übersicht (vgl. Abbildung 69) ersichtlich, ob weiterführende Dokumente für eine Steuer hinterlegt worden sind. Zudem wird das letzte Element der Versionshistorie angezeigt.

<sup>239</sup> Zur Vorgehensweise bei der Steuerartenrechnung vgl. Spengel, C., Steuerbelastungsvergleiche, 1995, S. 330, 353 f.

Abbildung 72: Einbezug einer Steuer in die HK ermöglichen und deren Aufteilung festlegen

**Distribution to production costs**

☒ Should this tax be included in the production costs?

Production in %

82

Administration in %

12

Sales in %

6

OK Abbrechen

Beim Verändern der Reihenfolge der Steuerberechnung wird geprüft, ob dies ohne Konflikte möglich ist oder nicht. Sollte dies nicht der Fall sein, wird eine Fehlermeldung angezeigt. Diese Reihenfolge ist wichtig, falls einzelne Steuern auf die Ergebnisse anderer Steuern in der aktuellen Periode zurückgreifen. Ebenso wird das Löschen einer Steuer verweigert, falls diese noch von anderen Modulen referenziert wird.

Mit dem „Mark as ...“-Button (vgl. Abbildung 69) kann der aktuelle Benutzer einen Eintrag in der Versionshistorie generieren und so eine Steuer z. B. als geprüft markieren.

Abbildung 73: Bemessungsgrundlage Grundsteuer

**Edit tax system**

Arithmetic operator	Name	Type
+	taxbase	Balance Sheet

**TaxBase**

Name: taxbase

Tax Base Balance Sheet Add

Edit TaxBase

Edit

Delete

change arithmetic operator

Rename

☒ Tax base?

Tarif: UniversalTarif Edit Tarif

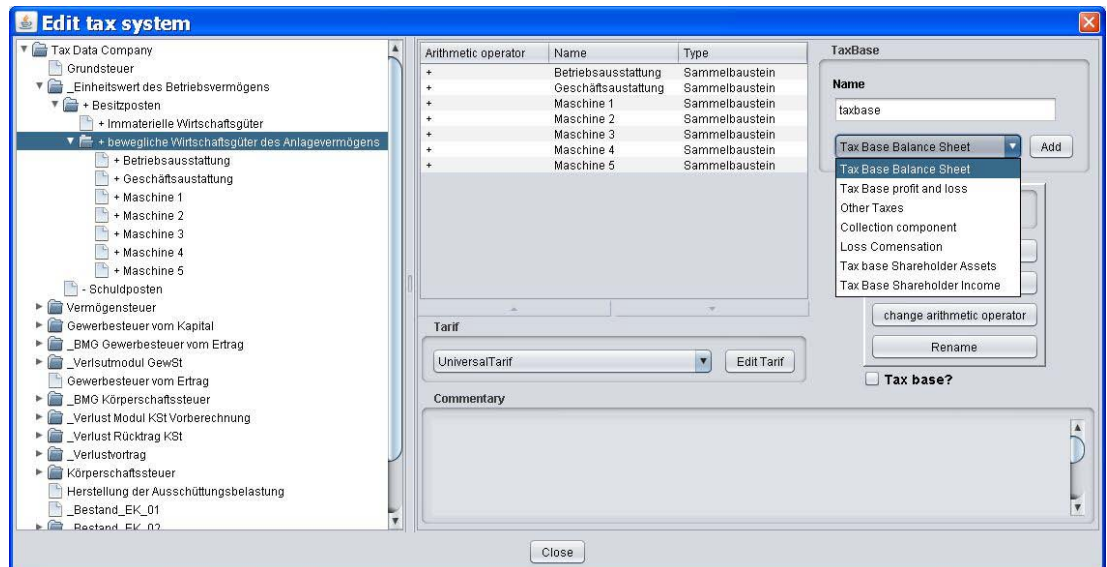
Commentary

Close

Jede Steuer enthält genau einen Sammelbaustein auf der obersten Ebene. Jede weitere Stufe kann beliebig viele dieser Bausteine enthalten und wird entsprechend im

Navigationsbaum auf der linken Seite (vgl. Abbildung 74) abgebildet. Das Verlustmodul enthält immer zwei dieser Elemente, eines für die Verluste oder vortragbaren Freibeträge und ein weiteres für die verwendbare Bemessungsgrundlage, sodass dieses Verlustmodul ebenfalls durch die Baumstruktur erfasst wird. Alle anderen Module stellen Blattknoten dar und werden nicht dargestellt.

**Abbildung 74: Ermittlung des Betriebsvermögens für die Vermögensteuer 1990**



Zur Ermittlung des Einheitswertes der Grundstücke für die Grundsteuer wird auf die Bilanz zurückgegriffen. Im konkreten Fall (vgl. Abbildung 75) werden die Anschaffungskosten mit  $0,3^{240}$  multipliziert. Hierbei ist jedoch zu beachten, dass Aktiva und Passiva zwei separate Teilbäume darstellen, sodass hier bei den Grundstücken und den Aktiva die Ergebnisse explizit durch das Einfügen von Formeln (i. d. R. reicht „x“ aus) nach oben geschleust werden müssen. Zum einen ist es dadurch möglich, Modifikationen für einzelne Gruppen auf dieser Ebene vorzunehmen. Bei der Vermögensteuer<sup>241</sup> genügt so z. B. die einmalige Multiplikation mit 140 %, anstatt diese Modifikation bei jedem Grundstück vorzunehmen. Zum anderen können so sehr einfach Teilbäume, z. B. zur Fehlerdiagnose ausgeschaltet werden, ohne alle Formeln entfernen zu müssen.

<sup>240</sup> „Die Grundstücke werden mit ihren jeweiligen Einheitswerten angesetzt, die pauschal als Prozentsatz der Anschaffungskosten eines Grundstückes bestimmt werden.“ Spengel, C., Steuerbelastungsvergleiche, 1995, S. 292.

<sup>241</sup> „Die ermittelten Einheitswerte werden jeweils auf volle 100 DM abgerundet, und mit 140 % angesetzt.“ Spengel, C., Steuerbelastungsvergleiche, 1995, S. 292.

Abbildung 75: Rückgriff auf Bilanzdaten zur Ermittlung des Einheitswertes für die Grundsteuer

Variablenname	Beschreibung
ahk oder ahk[0-9]	AHK der Periode: t-[0-9]
antBoden	Anteil der AHK diem Grund und Boden zurechen...
qmB[0-9]	Square meters of building (office or factory) buildi...
qmG[0-9]	Square meters of ground
rbw[0-9]	RBW Periode: t-[0-9]
t	aktuelle Periode
wbw oder wbw[0-9]	Wiederbeschaffungswert der Periode: t-[0-9]
xr	Wechselkurs
zwa oder zwa[0-9]	Zeitwert am Anfang der Periode: t-[0-9]
zwe oder zwe[0-9]	Zeitwert am Ende der Periode: t-[0-9]

Formel editieren

ahk \* 0.3

Check formular

Kommentar

Close

Schließlich muss bei der Grundsteuer noch der Tarif spezifiziert werden. „x“ ist der oben ermittelte Einheitswert aller Grundstücke. Dieser wird vor der Anwendung des Tarifs mit dem Wechselkurs multipliziert. Das Resultat wird wieder durch diesen geteilt. Dies ist notwendig, um Freibeträge und Tarifstufen in der konkreten Landeswährung abbilden zu können. Falls dies in seltenen Fällen nicht gewünscht sein sollte, kann diese Modifikation mithilfe der Variablen „xr“<sup>242</sup> rückgängig gemacht werden.

Die Tarifformel (vgl. Abbildung 76) wurde bewusst nicht ausmultipliziert, um so die Steuermesszahl (0,35 %) und den Grundsteuerhebesatz (1990: 351 %) zu trennen und zu verdeutlichen. Zudem hat sich in der beachteten Zeitreihe nur der Hebesatz geändert, sodass die Modifikationen für folgende Jahre in dieser Schreibweise einfacher durchzuführen sind.

<sup>242</sup> „xr“ repräsentiert den Wechselkurs für 1.000 € sodass dieser i. d. R. durch 1.000 zu teilen ist.

Abbildung 76: Eingabe des Tarifs der Grundsteuer

From (inclusive)	To (exclusive, if nextl...	Formula
0	9.223.372.036.854.7...	$x \cdot 0.0035 \cdot 3.51$

☐ Stufentarif (additive Stufen)  
☐ Inflationierung  
☐ in sich abzugsfähige Steuer

☒ Tatsächlicher Tarif der Steuer?

Commentary

Hebesatz 3.51. IFST-Schrift Nr. 301, Bonn 1991

Save Delete OK

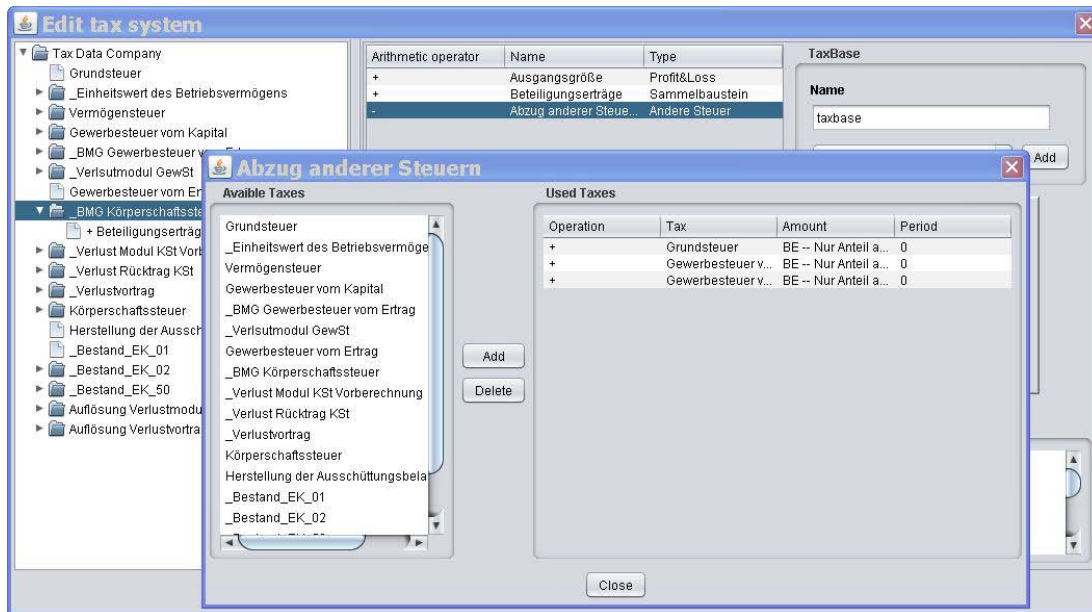
Manage PDF's

Open Add Delete

Durch die getroffenen Annahmen ist die Grundsteuer relativ einfach abzubilden und wurde deswegen auch als Beispiel ausgewählt. Für 1990 sind auf Unternehmensebene noch die Vermögensteuer, die Gewerbesteuern vom Kapital und vom Ertrag sowie die Körperschaftsteuer zu modellieren. Letztere erfordert aufgrund ihres gespaltenen Tarifs (einbehaltene Gewinne werden mit 50 %, während ausgeschüttete Gewinne mit 36 % besteuert werden) diverse Nebenrechnungen. Diese und die anderen Steuern sollen hier nicht im Detail dargestellt werden.

Zum Verständnis der weiteren Darstellungen späterer Jahrgänge seien auch die Tarife der anderen Steuern erwähnt. Der Tarif der Vermögensteuer für Kapitalgesellschaften beträgt 0.6 %. Der bundesdurchschnittliche Hebesatz der Gewerbesteuern für Gemeinden mit mehr als 50.000 Einwohnern, der den Standardberechnungen zugrunde liegt, betrug 1990 407 %.

Abbildung 77: Abzug anderer Steuern in der Körperschaftsteuer



Die Grundsteuer ist ebenso wie die Gewerbesteuer von der Bemessungsgrundlage der Körperschaftsteuer (vgl. Abbildung 77) abzugsfähig. Durch den Rückgriff auf die aktuelle Periode ist es wichtig, dass diese Steuern oberhalb der Körperschaftsteuer stehen und somit vor dieser berechnet werden. Sollte dies nicht der Fall sein, erscheint im Basisdialog eine entsprechende Fehlermeldung.

Wenn die referenzierte Steuer Bestandteil der Herstellungskosten ist, kann beim Rückgriff auch der Anteil (Betriebsergebnis,<sup>243</sup> Herstellungskosten oder komplett) spezifiziert werden, der berücksichtigt werden soll. Im konkreten Fall wird nur der Teil der Grundsteuer einbezogen, der nicht Teil der Herstellungskosten ist. Dieser beträgt aufgrund der funktionalen Aufteilung und der hinterlegten Formel 18 % (12% Verwaltung plus 6 % Vertrieb) der Steuerzahlung. Der andere Teil hat die Bemessungsgrundlage durch den Absatzvorgang<sup>244</sup> bereits gemindert oder wird diesen in Zukunft mindern. Somit würde eine Berücksichtigung der kompletten Steuerzahlung zu einem teilweisen doppelten Abzug führen.

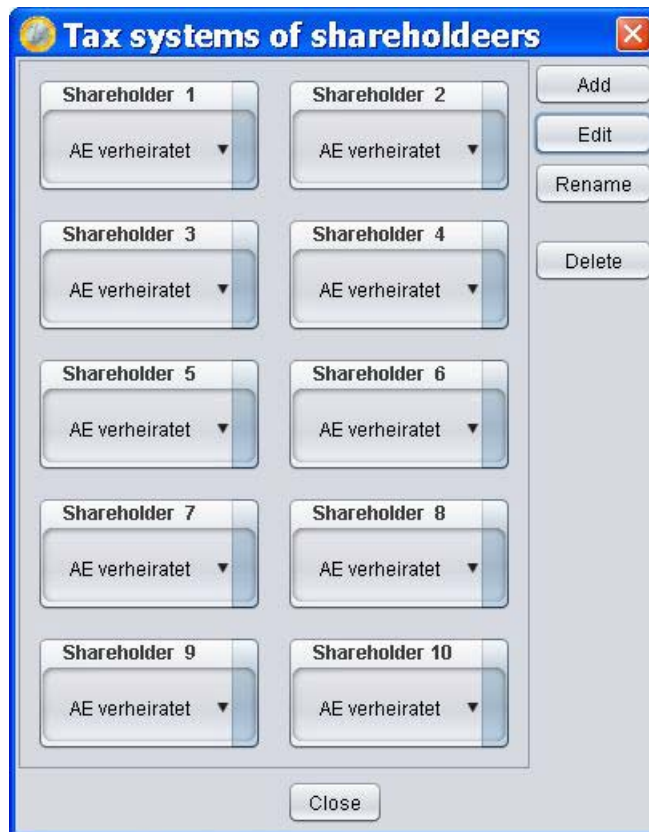
<sup>243</sup> Der Anteil, der nicht in die Ermittlung der Herstellungskosten einfließt. Vereinfachend wird dieser als Anteil, der in das Betriebsergebnis eingeht bezeichnet. „Komplett“ umfasst die gesamte Steuerzahlung. Der Standardwert ist BE für das Betriebsergebnis. Dieser entspricht der kompletten Steuerzahlung, wenn diese nicht Teil der Herstellungskostenermittlung ist.

<sup>244</sup> Die Herstellungskosten der abgesetzten Einheiten haben bereits die Bemessungsgrundlage gemindert. Der einbezogene Anteil der Grundsteuer ist Teil dieser Herstellungskosten. Werden Produkte eingelagert, wird dieser Aufwand in der Bilanz erfolgsneutral aktiviert. Erst bei der Veräußerung der Lagerbestände wird dieser Aufwand dann erfolgswirksam.



Für einen Vergleich der Gesamtsteuerbelastung sind außerdem die Steuersysteme für die Anteilseigner zu spezifizieren und zu modellieren. Im konkreten Fall wird angenommen, dass alle Anteilseigner verheiratet sind und den Splittingtarif in Anspruch nehmen.

*Abbildung 78: Zuordnung der Steuersysteme zu den Anteilseignern*



Auf Anteilseignerebene fällt zudem Vermögensteuer an. Aufgrund der unterschiedlichen Bewertungsmethoden<sup>245</sup> des Stuttgarter Verfahrens<sup>246</sup> waren vielfältige Nebenrechnungen notwendig. Des Weiteren stellt die Kirchensteuer einen Sonderfall dar, da sie von der Bemessungsgrundlage der Einkommensteuer abzugsfähig ist. Die festzusetzende Einkommensteuer ist aber gleichzeitig wiederum die Bemessungsgrundlage der Kirchensteuer, sodass dadurch eine zirkuläre Beziehung entsteht. Diese kann

<sup>245</sup> Maschinen können z. B. steuerlich degressiv abgeschrieben werden; das Stuttgarter Verfahren unterstellt jedoch einen linearen Werteverzehr.

<sup>246</sup> Das Stuttgarter-Verfahren (§ 11 Abs. 1 BewG [1990-1996] -- vgl. Abschnitt 76 ff. VStR 1989; Abschnitt 4 ff. VStR 1993; Abschnitt 4 ff. VStR 1995) ist zur Bewertung von nicht notierten Kapitalgesellschaftsanteilen notwendig. Es ermöglicht eine „Schätzung des gemeinen Wertes unter Berücksichtigung des Vermögens und der Ertragsaussichten.“ Spengel, C., et al., Substanzbesteuerung, 2010, S. 56.

nicht ohne Weiteres aufgelöst werden. Aufgrund der Annahme, dass „geschuldete und gezahlte Kirchensteuer der Periode (t) übereinstimmen“<sup>247</sup>, ist dies auch nicht notwendig. Der Grenzwert<sup>248</sup> des „iterativen Verfahrens“<sup>249</sup> entspricht unter Berücksichtigung der vorherigen Annahme einem in sich abzugsfähigen Tarif. Dadurch ist es auch möglich, den tatsächlichen Tarif zu verwenden. Es wird angenommen, dass die Anteilseigner in Bayern oder Baden-Württemberg wohnen und somit 8 %<sup>250</sup> Kirchensteuer zahlen müssen.

Abbildung 79: Kirchensteuertarif

**Bearbeite Tarif**

From (inclusive)	To (exclusive, if nextl...	Formula
-9.223.372.036.854....	9.223.372.036.854.7...	x * 0.08

☐ Stufentarif (additive Stufen)  
☐ Inflationierung  
☒ in sich abzugsfähige Steuer

☒ Tatsächlicher Tarif der Steuer?

**Commentary**

Steuersatz ergibt sich durch angenommene Selbstabzugsfähigkeit pauschal über die Formel  $8\% / (1 + 8\%)$  (Bayern und BW). In der neuen Version ist es nicht mehr notwendig, dies von Hand zu berechnen. Eine Differenzierung nach Bundesländern wird nicht vorgenommen.

Als Beispiel für einen nicht additiven Stufentarif kann der Tarif der Einkommensteuer (vgl. Abbildung 80) dienen. Allerdings handelt es sich genau genommen um einen additiven Tarif. In den Formeln der einzelnen Stufen wird jedoch die vorherige Ebene berücksichtigt. Bei einem Einkommen ab 120.042 DM wird z. B. die bei einem

<sup>247</sup> Spengel, C., Steuerbelastungsvergleiche, 1995, S. 306.

<sup>248</sup> Er entspricht dem Grenzwert der geometrischen Reihe und ist im Fall der Kirchensteuer gleich:  $0.08 / 1.08 = 0,074$ .

<sup>249</sup> Spengel, C., Steuerbelastungsvergleiche, 1995, S. 306.

<sup>250</sup> „Für die Kirchensteuer ist ein Steuersatz vorzugeben (Steuerdaten), der im Ausgangsfall 8 % beträgt.“ Spengel, C., Steuerbelastungsvergleiche, 1995, S. 306.



Tarif von 53 % zu viel berechnete Steuer in Höhe von 22.842 DM wieder abgezogen. Dies ist auch in den nachfolgenden Jahren<sup>251</sup> sehr gut erkennbar, da sich Änderungen der Grenzen und/oder Tarife auf diese absoluten Summen auswirken. Natürlich könnten die Tarifformeln so modifiziert werden, dass ein additiver Tarif genutzt werden könnte. Jedoch entspräche dieser dann nicht mehr dem kodifizierten Tarif. Deshalb erscheint dies nicht sinnvoll.

Abbildung 80: Einkommensteuertarif 1990

From (inclusive)	To (exclusive, if nextl...	Formula
-9.223.372.036.854....	5.617	0
5.617	8.154	$0.19 * x - 1067$
8.154	120.042	$(151.94 * ((x - 8100) / 1...)$
120.042	9.223.372.036.854.7...	$0.53 * x - 22842$

Buttons: Add Level, Remove last Level

☐ Stufentarif (additive Stufen)  
☐ Inflationierung  
☐ in sich abzugsfähige Steuer

☒ Tatsächlicher Tarif der Steuer?

Commentary

Est\_Tarif\_D1990 Save Delete OK

Bei der Überprüfung der einzelnen Steuern ist darauf zu achten, dass das aktuelle Vergleichsmodell evtl. nicht alle Zweige anspricht. Treten im aktuellen Rahmenmodell z. B. keine Verluste auf, werden die Regeln für die Verlustverrechnung nicht angesprochen. Daher empfiehlt es sich, verschiedene ökonomische Modelle zu testen.

Nachdem alle Parameter überprüft wurden, kann das Steuersystem als Bestandteil eines kompletten Modells (\*.tax) oder auch als eigenständige Datei (\*.ldt) jeweils als

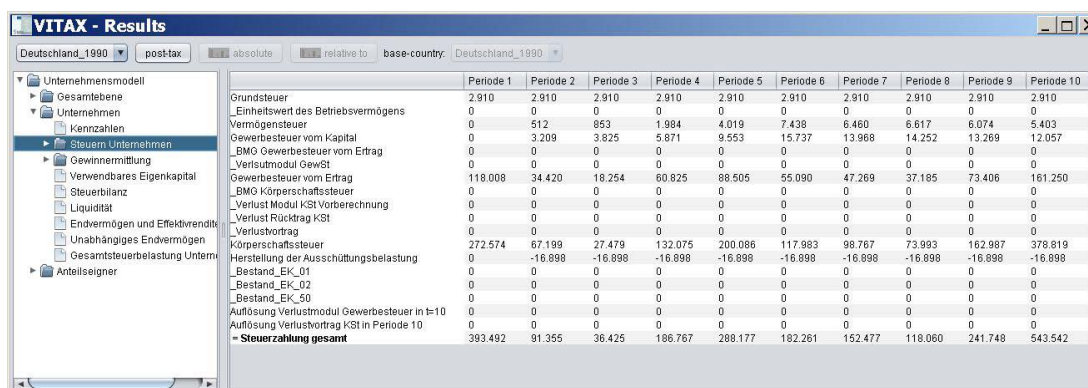
<sup>251</sup> Vgl. Spengel, C., et al., Substanzbesteuerung, 2010, S. 122 ff.

XML-Datei gespeichert werden. Diese Optionen sind aber nur für die temporäre Speicherung sinnvoll, daher wird die Speicherung in die Datenbank<sup>252</sup> benutzt.

Im Anschluss an die Modellierung und Kontrolle des ökonomischen Modells sowie des Steuersystems können die Steuerzahlungen (vgl. Abbildung 81) berechnet werden. Vermögensteuer und Gewerbesteuer vom Kapital fallen in der ersten Periode nicht an, da das ermittelte Betriebsvermögen negativ ist.

Eine weitere Besonderheit wird durch den gespaltenen Tarif bei der Körperschaftsteuer verursacht. Ausgeschüttete Gewinne werden niedriger besteuert (1990 mit 36 % statt 50 %) als einbehaltene. Zunächst werden alle Gewinne mit dem höheren Steuersatz belastet. Werden danach Teile des Gewinns ausgekehrt, erfolgt i. d. R. eine Steuerrückerstattung in Höhe der Differenz. Der genaue Ausschüttungsbetrag steht aber erst nach der Steuerberechnung fest, so dass die Rückerstattung der Differenz<sup>253</sup> erst in der nächsten Periode erfolgen kann. Die maximale Ausschüttung wird durch die Summe aus Gewinnrücklagen und Jahresüberschuss begrenzt, d. h. selbst wenn im Unternehmensplan eine fixe Ausschüttung vorgesehen ist, wird diese dann gegebenenfalls entsprechend gekürzt. Somit kann auch bei einem fest vorgegebenen Ausschüttungsbetrag dieser nicht direkt verrechnet werden. Um die verspätete Zahlung (Herstellung der Ausschüttungsbelastung) auszugleichen, wird diese entsprechend verzinst. Für Periode 10 muss dies auf Anteilseignerebene erfolgen, da auf dieser Ebene die Höhe der Ausschüttung dann feststeht.

*Abbildung 81: Steuern des Unternehmens für 1990 im Ausgangsfall*



	Periode 1	Periode 2	Periode 3	Periode 4	Periode 5	Periode 6	Periode 7	Periode 8	Periode 9	Periode 10
Grundsteuer	2.910	2.910	2.910	2.910	2.910	2.910	2.910	2.910	2.910	2.910
_Einheitswert des Betriebsvermögens	0	0	0	0	0	0	0	0	0	0
Vermögensteuer	0	512	853	1.984	4.019	7.438	6.460	6.617	6.074	5.403
Gewerbesteuer vom Kapital	0	3.209	3.825	5.871	9.553	15.737	13.968	14.252	13.269	12.057
_BMO Gewerbesteuer vom Ertrag	0	0	0	0	0	0	0	0	0	0
_Verlustmodul GewSt	0	0	0	0	0	0	0	0	0	0
Gewerbesteuer vom Ertrag	118.008	34.420	18.254	80.825	88.505	55.090	47.269	37.185	73.406	161.250
_BMO Körperschaftsteuer	0	0	0	0	0	0	0	0	0	0
_Verlust Modul KSt Vorberechnung	0	0	0	0	0	0	0	0	0	0
_Verlust Rücktrag KSt	0	0	0	0	0	0	0	0	0	0
_Verlustvortrag	0	0	0	0	0	0	0	0	0	0
Körperschaftsteuer	272.574	67.199	27.479	132.075	200.086	117.983	98.767	73.993	162.987	378.819
Herstellung der Ausschüttungsbelastung	0	-16.898	-16.898	-16.898	-16.898	-16.898	-16.898	-16.898	-16.898	-16.898
_Bestand_EK_01	0	0	0	0	0	0	0	0	0	0
_Bestand_EK_02	0	0	0	0	0	0	0	0	0	0
_Bestand_EK_60	0	0	0	0	0	0	0	0	0	0
Auflösung Verlustmodul Gewerbesteuer in t=10	0	0	0	0	0	0	0	0	0	0
Auflösung Verlustvortrag KSt in Periode 10	0	0	0	0	0	0	0	0	0	0
<b>= Steuerzahlung gesamt</b>	<b>393.492</b>	<b>91.355</b>	<b>36.425</b>	<b>186.767</b>	<b>288.177</b>	<b>182.261</b>	<b>152.477</b>	<b>118.060</b>	<b>241.748</b>	<b>543.542</b>

<sup>252</sup> Dies geschieht für den Benutzer transparent und erfordert keine besonderen Kenntnisse. Eine detaillierte Beschreibung ist in Kapitel 6.7 zu finden.

<sup>253</sup> Einbehaltene Gewinne werden mit 50 % besteuert. Ausgeschüttete aber nur mit 36 %, sodass 14 % zu viel an Körperschaftsteuer bezahlt wurden.

Abbildung 82 zeigt die aggregierten Steuerzahlungen der Anteilseigner. Im konkreten Fall erhalten die Anteilseigner 3-10 in Periode 1 und 2 eine Steuerrückerstattung (negative Steuer), da die anrechenbare Körperschaftsteuer auf die ausgeschüttete Dividende höher ist als die auf die Summe der Einkünfte zu zahlende Einkommenssteuer.<sup>254</sup>

**Abbildung 82: Steuern der Anteilseigner für 1990 im Ausgangsfall**

	period 1	period 2	period 3	period 4	period 5	period 6	period 7	period 8	period 9	period 10
shareholder 1	11.353	12.471	13.615	16.731	18.595	21.655	22.431	23.298	23.867	202.200
shareholder 2	133	303	482	1.008	1.317	1.838	1.959	2.096	2.181	24.976
shareholder 3	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 4	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 5	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 6	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 7	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 8	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 9	-91	-25	44	251	425	716	785	864	913	7.925
shareholder 10	-91	-25	44	251	425	716	785	864	913	7.925

## 6.5 Folgende Jahre

Der erste Jahrgang dient als Ausgangsbasis für die folgenden Jahre. Neben reinen Tarifänderungen sind Änderungen an den Bemessungsgrundlagen berücksichtigen. Des Weiteren ist für die Folgejahre auf das Aussetzen oder die Abschaffung einzelner Steuern oder deren Neueinführung zu achten.

Nachdem 1990 in der Datenbank hinterlegt<sup>255</sup> wurde oder alternativ temporär als XML-Datei (\*.ldt) abgespeichert wurde, kann nun dieser Jahrgang erneut geladen oder direkt via „Change Country“ (vgl. Abbildung 83) geändert werden. In beiden Fällen erscheint der Dialog aus Abbildung 83. Im konkreten Fall genügt es, die Jahreszahl zu erhöhen. Danach müssen die Änderungen (vgl. Tabelle 4), wie im Falle von 1990 (vgl. 6.4), entsprechend modelliert und gespeichert werden. Dieser Vorgang wiederholt sich für jedes Jahr. Auf diese Weise entsteht eine Zeitreihe von 1990 bis 2009.

<sup>254</sup> Der Durchschnittssteuersatz dieser Anteilseigner liegt unter 36 %.

<sup>255</sup> Für den Nutzer erfolgt dies transparent, da dies ebenso wie die normale Speicherung über einen Menüpunkt erfolgt. Anstelle der Eingabe eines Dateinamens wird nach einem Kommentar gefragt und gegebenenfalls, ob das Land überschrieben und/oder validiert werden soll. Das Laden erfolgt über den Dialog aus Abbildung 88.

Abbildung 83: Landänderungsdialog nach erneutem Laden

**Land exists already!**

Name: Deutschland

Year: 1.990

Variation: CURRENT\_LAW

Commentary of Variation:

Displayname: Deutschland\_1990

Exchange rate to 1000€: 1.956

Symbol of the national currency: DM

☒ Use base-pretax-model

Change Close

Tabelle 4 listet die berücksichtigten Steuerrechtsänderungen<sup>256</sup> auf. Bei Jahrgängen ohne Kommentierung gab es keine relevanten Änderungen.

Tabelle 4. Auflistung der relevanten Steuerrechtsänderungen von 1991-2009

Jahr	Änderungen
1991	<p>Die LIFO-Methode wird für die Vermögensteuer übernommen.</p> <p>Erhebung eines Solidaritätszuschlages in Höhe von 7,5% auf die Körperschaftsteuer und die Einkommensteuer. Dieser gilt jedoch nur für den Zeitraum von 01.07.1991 bis zum 30.06.1992. Im Modell findet keine unterjährige Gewinnermittlung statt, sodass dies durch den halben Tarif von 3,75 % abgebildet wurde. Dieser wird auch bei den Anteilseignern erhoben.</p>
1992	<p>Lokale Begünstigungen, wie z. B. die Aussetzung der Vermögensteuer und der Gewerbesteuer vom Kapital in den neuen Bundesländern werden nicht berücksichtigt.</p> <p>Solidaritätszuschlag wird wie 1991 nur mit dem halben Tarifsatz abgebildet.</p>

<sup>256</sup> Für die vollständige Liste siehe Spengel, C., et al., Substanzbesteuerung, 2010, S. 108-120.

Jahr	Änderungen
1993	Übernahme der Steuerbilanzwerte in die Vermögensaufstellung (§ 109 BewG)
	Erhöhung des Freibetrags für Betriebsvermögen im Rahmen der Vermögensteuer auf 500.000 DM
	Erhöhung des Sparerfreibetrages auf 6.000 DM
1994	
1995	Wiedereinführung des Solidaritätszuschlages in Höhe von 7,5%.
	Anhebung des Vermögensteuersatzes für Grundvermögen und sonstiges Vermögen mit Ausnahme der Beteiligungswerte auf 1 % ab 1.1.1995 (§ 10 Nr. 1 VStG)
1996	
1997	Die Vermögensteuer wird nicht mehr erhoben.
1998	Die Gewerbesteuer vom Kapital wird abgeschafft.
	Senkung des Solidaritätszuschlages auf 5,5%.
1999	Begrenzung des Verlustrücktrags auf ein Jahr und eine Million DM. Die temporäre Begrenzung auf 2 Millionen wird nicht berücksichtigt.
2000	Reduktion des Sparerfreibetrags auf 3.000 DM.
2001	Abschaffung der Tarifspaltung bei der Körperschaftsteuer. Ausgeschüttete und thesaurierte Gewinne werden nun gleich besteuert.
	Einführung des Halbeinkünfteverfahrens für Dividenden.
	Degressive Abschreibung ist nur noch mit max. 20 % statt bisher 30 % möglich.
	Erhöhung der Abschreibungsdauer von Gebäuden im Betriebsvermögen von 25 auf 33 Jahre.
	Anpassung der AFA-Tabellen. Dadurch ändern sich die Abschreibungsdauern der Maschinen im Modell.
2002	Mindestalter zur steuerlichen Ansetzung von Verpflichtungen aus der betrieblichen Altersversorgung sinkt von 30 auf 28 Jahre.
	Der Einkommensteuertarif lautet nun auf Euro, daher beträgt der Wechselkurs nun 1000 statt 1956.

Jahr	Änderungen
<b>2003</b>	Der Körperschaftsteuersatz wird durch das Flutopfersolidaritätsgesetz temporär auf 26,5 % erhöht.
<b>2004</b>	<p>Einführung der sog. Mindestbesteuerung für die Körperschaftsteuer, Gewerbesteuer und die Einkommensteuer, indem der Verlustvortrag auf eine Million plus 60 % des eine Million übersteigenden Gewinns beschränkt wird.</p> <p>Hinzurechnung von 5 % des Gewinns aus inländischen Beteiligungserträgen im Sinne von § 8b Abs. 1 und 2 KStG als nicht abzugsfähige Betriebsausgabe (§ 8b Abs. 3 und 5 KStG)</p> <p>Sparerfreibetrag wird auf 1.370 € reduziert.</p>
<b>2005</b>	
<b>2006</b>	Erhöhung des maximalen degressiven Abschreibungssatzes auf 30 %. Dieser darf jedoch nicht mehr als das Dreifache der linearen Abschreibung betragen.
<b>2007</b>	Kürzung des Sparerfreibetrags auf 750 €.
<b>2008</b>	<p>„Wegfall der 50 % Hinzurechnungen von Dauerschulden bei der Gewerbesteuer sowie hinzuzurechnender Mieten und Pachten (§ 8 GewStG a.F.)“</p> <p>25% Hinzurechnung von sämtlichen Finanzierungsaufwendungen bei der Gewerbesteuer mit einem Freibetrag von EUR 100.000 (§ 8 Nr. 1 GewStG)</p> <p>Abschaffung der degressiven Abschreibungsmöglichkeiten für bewegliche Wirtschaftsgüter des Anlagevermögens (§ 7 Abs. 2 EStG)</p> <p>Einführung der Zinsschranke</p>

Jahr	Änderungen
2009	Absenkung des steuerlichen Mindestalters auf 27 Jahre im Rahmen der Bildung von Pensionsrückstellungen (§ 6a EStG)
	Wiedereinführung der degressiven Abschreibung für bewegliche Wirtschaftsgüter des Anlagevermögens in Höhe von 25 % befristet auf zwei Jahre (§ 7 Abs. 2 EStG)
	Zeitlich befristete Anhebung der Freigrenze bei der Zinsschranke auf EUR 3.000.000 (§ 4 h Abs. 2 EStG)
	Einführung der Abgeltungsteuer in Höhe von 25% auf Kapitalerträge mit Veranlagungsoption für wesentliche Beteiligungen (§ 20 und § 32d EStG)

Die reinen Tarifanpassungen sollen hier nicht explizit dargelegt werden, da sie bei der Recherche im Webinterface (vgl. 6.9.2) dargestellt werden. Im Gegensatz zur Studie für die Industrie und Handelskammer<sup>257</sup> werden hier nur die für eine Kapitalgesellschaft und deren Anteilseigner relevanten Änderungen aufgeführt, die zur Umsetzung im Modell benötigt werden.

Eine deutliche Vereinfachung bei der Steuermodellierung ist ab 2001 möglich, da zu diesem Zeitpunkt die Tarifspreizung für einbehaltene und ausgeschüttete Gewinne bei der Körperschaftsteuer aufgehoben wurde und seitdem ein einheitlicher Tarif gilt. Dadurch können viele Hilfsrechnungen, wie z. B. für die Herstellung der Ausschüttungsbelastung, entfallen. Auch die Verlustverrechnung wird deutlich einfacher.

## 6.6 Dokumentation hinzufügen

Nachdem nun die Zeitreihe komplettiert wurde, muss noch die Dokumentation hinzugefügt werden. Prinzipiell sollte diese schon während der Modellierung (vgl. 6.4) erfolgen. Sie wird hier nur aus didaktischen Gründen nachgeholt. Folgende Dokumente sollten wenn möglich, für jedes Jahr hinterlegt werden:

- Gesetzestexte, Richtlinien, Kommentare und Urteile;

---

<sup>257</sup> Vgl. Spengel, C., et al., Substanzbesteuerung, 2010, S. 108-120.

- Statistische Angaben (wie z. B. der Durchschnittshebesatz für die Gewerbesteuer für Gemeinden mit mehr als 50.000 Einwohner);
- Weiterführende Literatur (z. B. Erläuterungen zu Bewertungsverfahren);
- Erläuterungen über getroffene Annahmen (z. B. in welche Kategorie eine Maschine eingeordnet wurde).

Insbesondere sollten auch alle Gesetzestexte eingescannt und hinterlegt werden, da es sich als recht schwierig erwiesen hat, diese für vergangene Jahre zu recherchieren. Übersichten über Regelungen im Zeitablauf sollten zumindest bei jedem Jahr der Änderung hinterlegt werden. Die damit einhergehende Redundanz doppelter Dokumente ist nicht optimal, jedoch zugunsten des vereinfachten Auffindens relevanter Dokumente in Kauf zu nehmen. Es wäre zwar möglich, doppelte Dokumente zu Erkennen<sup>258</sup> und in der Datenbank nur einmal zu speichern. Hierdurch würde dann aus der 1:n Beziehung zwischen Land und PDF-Datei eine n:m Beziehung. Dies würde im Datenbankschema eine separate Verknüpfungstabelle erfordern und dieses weiter verkomplizieren. Daher wird dieser eher seltene Fall nicht explizit behandelt und die Duplikation akzeptiert. Alternativ können die Dokumente evtl. auch aufgeteilt<sup>259</sup> werden.

**Abbildung 84: Erkennung vorhandener weiterführender Dokumentation auf Länderebene**



<sup>258</sup> Hier zu müsste der komplette physische Inhalt (Byte für Byte) verglichen werden, da der Dateiname keine Eindeutigkeit garantiert.

<sup>259</sup> Sind z. B. die Regelungen für 1990 auf Seite 1 hinterlegt und für das nächste Jahr auf Seite 2. Könnte die erste Seite bei 1990 hinterlegt werden und die zweite bei 1991.



Durch den aktivierten „Open“-Button (vgl. Abbildung 84) ist ersichtlich, dass neben normalen Textkommentaren auch weiterführende PDF-Dokumente hinterlegt sind.

Auf der Ebene der Steuern ist dann die Checkbox (vgl. Abbildung 86) entsprechend gesetzt. Durch einen Doppelklick erscheint ein Auswahldialog, der dann das gewünschte Dokument in der dem Dateiformat zugeordneten Anwendung<sup>260</sup> anzeigt. Das Löschen und das Hinzufügen von weiteren Dokumenten erfolgt auf Tarifebene (vgl. Abbildung 85) oder im Hauptmenü auf Länderebene (vgl. Abbildung 84).

*Abbildung 85: Tarif der Abgeltungsteuer*

From (inclusive)	To (exclusive, if nextl...	Formula
-9.223.372.036.854....	1.603	0
1.603	9.223.372.036.854.7...	$(x - 1602) / 4,08$

Buttons: Add Level, Remove last Level

☐ Stufentarif (additive Stufen)  
☐ Inflationierung  
☐ in sich abzugsfähige Steuer

☒ Tatsächlicher Tarif der Steuer?

**Commentary**

$e - 4q / (4 + k) K == 0.08; q = 0 ==> e / 4,08$

Abgeltungssteuertarif 2009 [v] Save Delete OK

Regelungen zu einzelnen Steuern sollten im Tarif hinterlegt werden. Vorschriften bzgl. Abschreibungen, betrieblicher Altersvorsorge, Herstellungskosten und Verbrauchsfolge sollten im jeweiligen Land hinterlegt werden. Diese Bestimmungen haben unmittelbaren Einfluss auf die Gewinnermittlung. Sie sind im Metamodell für

<sup>260</sup> In den meisten Fällen wird dies der „Acrobat Reader“ sein. Die Anwendung benutzt aber die für diesen Dateityp im Betriebssystem registrierte Anwendung.

die Steuermodellierung aber nur indirekt<sup>261</sup> zugreifbar. Daher ist es wenig sinnvoll, diese Bestimmungen einer einzelnen Steuer zuordnen, obwohl sie Teil der Bemessungsgrundlage sind.

Auf der Länderebene können alle Dokumente eingesehen und verwaltet werden. Dadurch wirkt sich das Löschen eines Tarifs nicht auf die hinterlegten Dokumente aus. Jedoch müssen nun nicht mehr benötigte Dokumente auf Länderebene explizit gelöscht werden. Der potentielle Schaden durch versehentliches Löschen dürfte jedoch größer sein als der Mehraufwand des expliziten Löschens, sodass diese Vorgehensweise gewählt wurde.

Diese Verwaltung (Anzeigen, Speichern und Löschen von PDF-Dokumenten) ist auch im Webinterface (vgl. Abbildung 87) möglich. Sie stellt dort bisher die einzige Möglichkeit dar, die Informationen bzgl. eines Landes zu verändern.

**Abbildung 86: Erkennung und Auswahl vorhandener weiterführender Dokumentation auf Steuerebene**

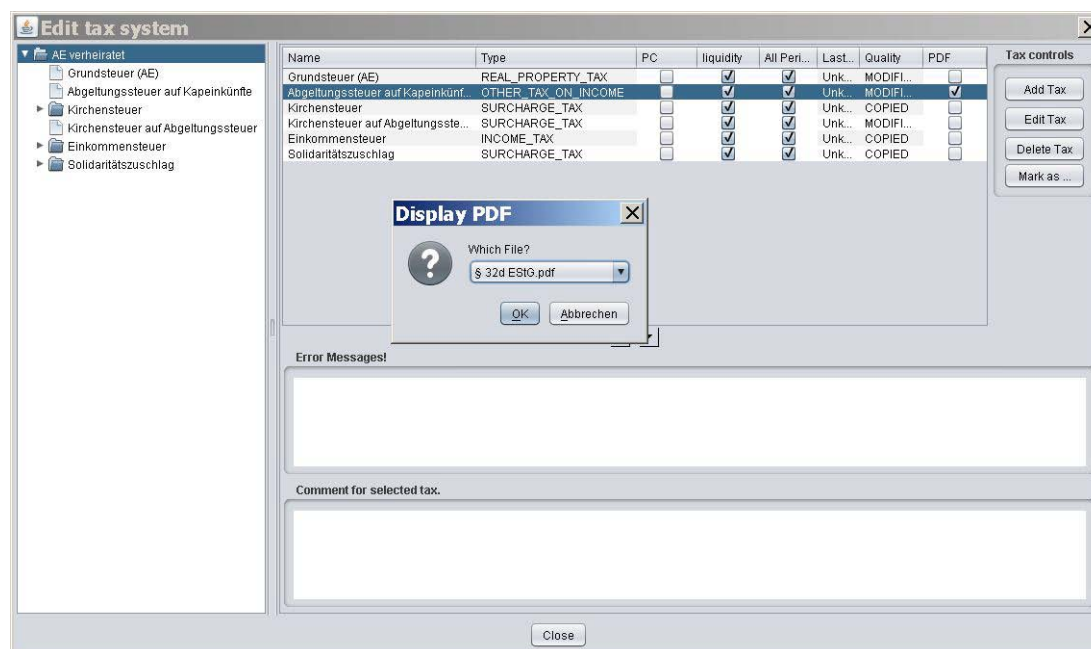


Abbildung 87 zeigt einen Überblick über die Abgeltungsteuer für Deutschland 2009. Sie wurde als *OTHER\_TAX\_ON\_INCOME* klassifiziert. Des Weiteren wird sie für alle Perioden berechnet und ist liquiditätswirksam. Die einzige Bemessungsgrundla-

<sup>261</sup> Indirekt bedeutet, dass die Resultate dieser Regelung verfügbar sind, aber nicht die Parameter, die dazu geführt haben. Bei Wirtschaftsgütern ist z. B. der Restbuchwert verfügbar. Ebenso kann auf den resultierenden Gewinn vor Steuern zugegriffen werden.

ge ist vom Typ *BMG\_Shareholder\_Asset*. Unter diesem Punkt sieht man den aktuellen Tarif. Die erste Stufe spiegelt den Sparerpauschbetrag in Höhe von 1602 € für Ehepaare wider. Danach wird der 1602 € übersteigende Betrag durch  $4.08^{262}$  geteilt. Die Datei zur Dokumentation heißt „§ 32d EStG.pdf“ und enthält den eingescannten Gesetzestext. Darunter ist noch die Historie zu sehen. Abbildung 85 zeigt den Tarif und die Bemessungsgrundlage in der Desktop-Version. Der aktivierte „Open“-Button weist auf die hinterlegte Dokumentation hin.

Abbildung 87: Verwaltung von Dokumenten im Webinterface

The screenshot shows the Vitax Webinterface. The left sidebar contains a navigation menu with the following items: Germany, 2009\_CURRENT\_LAW, PDFs, Betriebliche, Altersvorsorge, Abschreibungen, Herstellungskosten, Steuerdaten, Anteilseigner, Steuerdaten, AE verheiratet, Grundsteuer (AE), Abgeltungssteuer auf Kapeinkünfte (selected), Dividenden und Zinsen, Kirchensteuer, Kirchensteuer auf Abgeltungssteuer, Einkommensteuer, and Solidaritätszuschlag. The main content area is titled 'Abgeltungssteuer auf Kapeinkünfte'. It includes a 'Typ' dropdown set to 'OTHER\_TAX\_ON\_INCOME', checkboxes for 'In HK:', 'Liquidationswirksam:', and a 'Kommentar:' field. Below this is the 'Herstellungskosten' section with checkboxes for 'Einbezogen:' and 'Einbezogen in AS:'. The 'Steuer in Periode berechnet' section shows a table with 10 columns (1-10) and 10 rows (1-10), all containing a checkmark. The 'Enthaltene Bemessungsgrundlagen' section contains a table with columns 'Operation', 'Name', and 'Typ'. The 'Universaler Tarif - Abgeltungssteuertarif 2009' section contains a table with columns 'Von', 'Bis', and 'Funktion'. The 'PDF Dateien' section contains a table with columns 'Datei Name' and 'Kommentar', and buttons for 'Anzeigen', 'Löschen', and 'Hinzufügen'. The 'Historie' section contains a table with columns 'Nutzer', 'Qualität', 'Datum & Zeit', and 'Herkunftsland'.

Operation	Name	Typ
+	Dividenden und Zinsen	BMG_Shareholder_Asset

Von	Bis	Funktion
-9223372036854775808	1603	0
1603	9223372036854775807	(x -1602) / 4.08

Datei Name	Kommentar
§ 32d EStG.pdf	

Nutzer	Qualität	Datum & Zeit	Herkunftsland
Unknown	NEW	12/10/09 12:36 PM	

<sup>262</sup> In der Formel  $\frac{e-4q}{4+k}$  (§32d EStG Abs. 1 Nr. 4) wird „k“ durch 0.08 ersetzt und entspricht dem Kirchensteuersatz für Bayern und Baden-Württemberg. Der Zähler „e-4q“ verkürzt sich zu „e“ (= Einkünfte nach § 20 EStG), da es keine anrechenbare ausländische Steuer (q) gibt.

## 6.7 Speichern und Laden

Der Auswahldialog zum Laden von Ländern aus der Datenbank (vgl. Abbildung 88) ist eine der wenigen Stellen, an denen die Desktop-Anwendung direkte Abfragen benutzt, um so nur die ID eines Landes aus der Datenbank zu extrahieren. Andernfalls würden alle Länder sofort geladen, unabhängig davon, ob sie später auch ausgewählt werden. Dies hätte zur Folge, dass fast die komplette Datenbank geladen werden würde. Somit ist hier eine direkte Abfrage zwingend notwendig. Mithilfe der ID werden dann die ausgewählten Länder aus der Datenbank geladen.

*Abbildung 88: Auswahldialog zum Laden aus der Datenbank*



Abbildung 88 zeigt alle in der Datenbank zurzeit hinterlegten Länder nach Jahren sortiert. Aus Gründen der Übersichtlichkeit sind zunächst nur die Jahre sichtbar. Zur Auswahl muss das gewünschte Jahr (z. B. 2007 und 2008) expandiert und anschließend das anvisierte Land selektiert werden. Es können auch beliebig viele Länder durch Mehrfachauswahl gleichzeitig geladen werden.

Abbildung 89: Simulationsmenü



Nachdem sich der Benutzer angemeldet<sup>263</sup> hat (vgl. 6.2), kann er auch neue oder veränderte Länder in der Datenbank speichern (vgl. „Save Country to DB“ - Abbildung 89) oder löschen. Beim Löschen kann man entweder wie beim Laden aus einem vergleichbaren Auswahldialog das gewünschte Land selektieren oder das in der Desktopanwendung aktuell ausgewählte direkt entfernen.

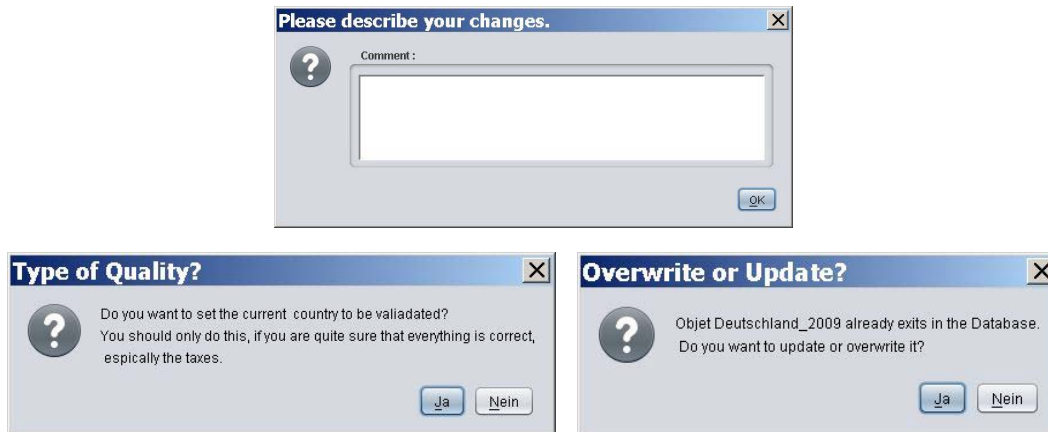
Abbildung 90: Datenbankmenü



Je nach Auswahl der Optionen im Datenbankmenü (vgl. Abbildung 90) werden Nachfragen bzgl. des Überschreibens (vgl. Abbildung 91) bereits existierender Länder oder deren Validierung beim Speichern eingeblendet. Die Nachfrage bezüglich eines beschreibenden Kommentars der vorgenommen Änderungen wird immer eingeblendet. Will man Änderungen an mehreren Ländern persistieren, kann man alle zu diesem Zeitpunkt geladenen Länder via „Commit all changes“ speichern.

<sup>263</sup> Die Rolle für den ausschließlich lesenden Zugriff wird zurzeit nicht verwendet; daher hat jeder Benutzer nach der Anmeldung automatisch Schreibrechte.

**Abbildung 91: Dialogabfolge beim Speichern in die Datenbank**



Es besteht zudem die Möglichkeit der Speicherung als Datenbank abhängiges Modell (vgl. Abbildung 89), d. h. es werden nur das aktuelle ökonomische Modell sowie die Schlüsselattribute der jeweiligen Länder gespeichert. Beim Öffnen werden dann die Länder mithilfe dieser Attribute aus der Datenbank nachgeladen. Somit besteht die Möglichkeit, automatisch auf die jeweils aktuellen Modellierungen zurückzugreifen, ohne diese explizit per Hand aus der Datenbank nachladen zu müssen.

Wenn am Ende eines Projektes der aktuelle Stand archiviert werden soll, um jederzeit die Ergebnisse reproduzieren zu können, sollte dieser Rückgriff nicht verwendet werden, da sich ansonsten auch zukünftige Änderungen auswirken. Hierfür sollte ein komplettes Modell als XML-Datei gespeichert werden. Aufgrund des temporären Charakters kann diese Speicherung nicht zur Langzeitarchivierung empfohlen werden, da die Daten durch Änderungen am Programmcode unbrauchbar werden könnten. Zur Archivierung von Projekten sollte daher auch immer die verwendete Programmversion mit archiviert werden.

Bei unverändertem Datenbankschema sollte es dann auch möglich sein, das alte Modell über die Datenbank zu konvertieren. Dazu würde es in einer alten Version geladen und in der Datenbank gespeichert werden, um anschließend mit der aktuellen Programmversion wieder geladen und erneut gespeichert zu werden. Dies ist jedoch ungewiss, sodass nur die Datenbank zur längerfristigen Speicherung benutzt werden sollte. Werden solche Projektdaten längerfristig benötigt, sollten daher entsprechende Variationen in der Datenbank angelegt werden.

Prinzipiell wird das ökonomische Rahmenmodell nur lokal gespeichert. Für temporäre Zwecke ist es auch möglich, gezielt einzelne Länder abzuspeichern und zu laden.



Zudem existiert die Option, ein Land nur aus dem aktuell bearbeiteten Modell, nicht aber aus der Datenbank zu entfernen.

## 6.8 Berechnen einer Zeitreihe

Nachdem die Jahre 1990-2009 (vgl. 6.4 und 6.5) in die Datenbank eingepflegt worden sind, kann eine Zeitreihe berechnet werden. Der ökonomische Rahmen entspricht einem mittelständischen Unternehmen des Verarbeitenden Gewerbes. Aus empirischen Studien der Bundesbank<sup>264</sup> wurden hierfür Daten bzgl. der Vermögens- und Kapitalstruktur sowie Bilanz- und Erfolgskennzahlen<sup>265</sup> ermittelt.

Tabelle 5: Steuerbilanz des Unternehmens in Periode 6

AKTIVA		EUR	PASSIVA		EUR
A.	Anlagevermögen		A.	Eigenkapital	
I.	Immaterielle Vermögensgegenstände	16.972	I.	Einlagen	350.000
II.	Sachanlagevermögen		II.	Gewinnrücklagen	739.301
1.	Grundstücke	925.126	III.	Jahresüberschuss/ Jahresfehlbetrag	209.216
2.	Maschinen	697.857			
3.	Betriebs- und Geschäftsausstattung	39.675	B.	Rückstellungen	
III.	Finanzanlagen		I.	Pensionen	408.874
1.	Beteiligungen	40.000	II.	Sonstige	0
2.	Langfristige Forderungen	30.000	C.	Verbindlichkeiten	
B.	Umlaufvermögen		I.	Verbindlichkeiten gegenüber Dritten	550.000
I.	Vorräte	1.538.976	II.	Verbindlichkeiten gegenüber Gesellschaftern	720.000
II.	Forderungen aus Lieferung und Leistung	1.453.156	III.	Verbindlichkeiten aus Lieferung und Leistung	836.445
III.	Wertpapiere	408.874	IV.	Kurzfristige Verbindlichkeiten	2.160.000
IV.	Kasse, Guthaben	823.183			
SUMME		5.973.819	SUMME		5.973.819

Quelle: Spengel, C. et al., Substanzbesteuerung, 2010, S. 64.

Fehlende Daten (wie z. B. der Produktions- und Absatzplan) wurden sinnvoll ergänzt, sodass eine größtmögliche Annäherung an diese Referenzzahlen gewährleistet ist und somit ein realistisches Szenario entsteht.

Tabelle 6: Kennzahlen der Modellunternehmen

	Anlageintensität	Umsatzrentabilität	Eigenkapitalrentabilität	Eigenkapitalquote	Vorratsintensität	Personalintensität
Verarbeitendes Gewerbe (mittelständisches Modellunternehmen)	27,83%	2,59%	19,21%	21,74%	25,76%	29,71%

Quelle: Spengel, C. et al., Substanzbesteuerung, 2010, S. 64.

<sup>264</sup> Vgl. Deutsche Bundesbank, 2003, S. 12-168.

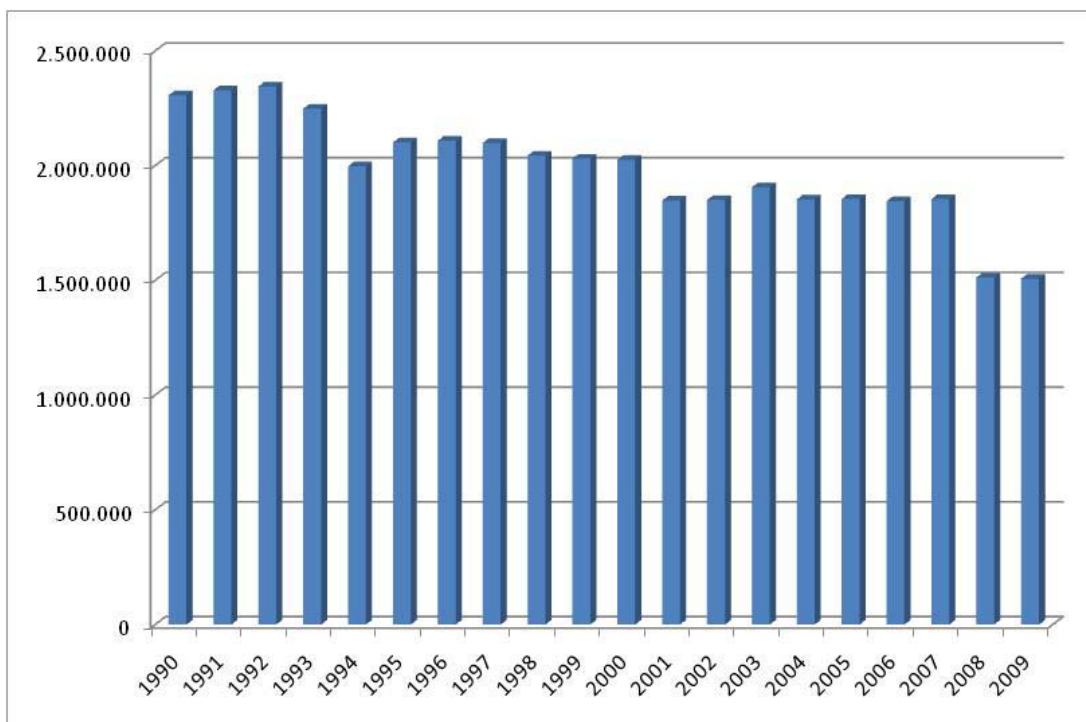
<sup>265</sup> Vgl. Spengel, C., et al., Substanzbesteuerung, 2010, S. 63 ff.

Das Unternehmen besitzt 10 Anteilseigner. Alle Anteilseigner sind verheiratet und haben keine Kinder. Ebenso besitzen alle ein Vermögen von 25.000 €, das nicht inflationiert wird, und virtuelle Einkünfte in gleicher Höhe. Die Anteile an der Gesellschaft entsprechen auch dem Anteil am Gesellschafterdarlehn und betragen für den ersten Anteilseiger 51 %, für den zweiten 9 % und für die restlichen 5 %.

Die Steuerbelastung in Periode 10 ergibt sich aus dem Vergleich des Endvermögens des Unternehmens, das ohne und mit Steuern veranlagt wurde. Alle stille Lasten und Reserven wurde gehoben und die restlichen Verlustvorträge bewertet. In der letzten Periode schüttet das Unternehmen seine kompletten Gewinnrücklagen inkl. des aktuellen Jahresüberschusses und der oben genannten Modifikationen aus.

Somit kann die Steuerbelastung auf Anteilseignerebene steigen, obwohl sich die steuerlichen Rahmenbedingungen verbessert haben, da die verzinste<sup>266</sup> Steuerersparnis in der letzten Periode ausgekehrt und versteuert wird.

*Abbildung 92: Steuerbelastung in Deutschland auf Unternehmensebene im Zeitablauf*



In der Zeitreihe auf Unternehmensebene lassen sich die Veränderungen beim Körperschaftsteuersatz sehr gut ablesen. So markieren die Senkungen größere Sprünge,

<sup>266</sup> Musste das Unternehmen zur Zahlung der Steuer vorher einen Kredit aufnehmen, kann dies auch eingesparte Zinszahlungen beinhalten.



wie z. B. im Jahr 1994, als der Körperschaftsteuersatz von 50 % auf 45 % gesenkt wurde. Ebenso sind die weiteren Tarifreduzierungen (1999 auf 40 %, 2001 auf 25 % und die Senkung auf 15 % im Jahre 2008) gut erkennbar und erklären auch den größten Teil der Änderungen.

1991 erfolgt ein leichter Anstieg durch die halbjährige Einführung des Solidaritätszuschlages. Die Hebesätze für die Gewerbesteuer und Grundsteuer waren in diesem Jahr niedriger als 1992, sodass hierdurch auch die Gesamtbelastung des Unternehmens geringfügig niedriger ist. 1993 wirken sich primär die Veränderungen bei der Vermögensteuer (Übernahme der Steuerbilanzwerte und erhöhter Freibetrag) belastungsmindernd aus.

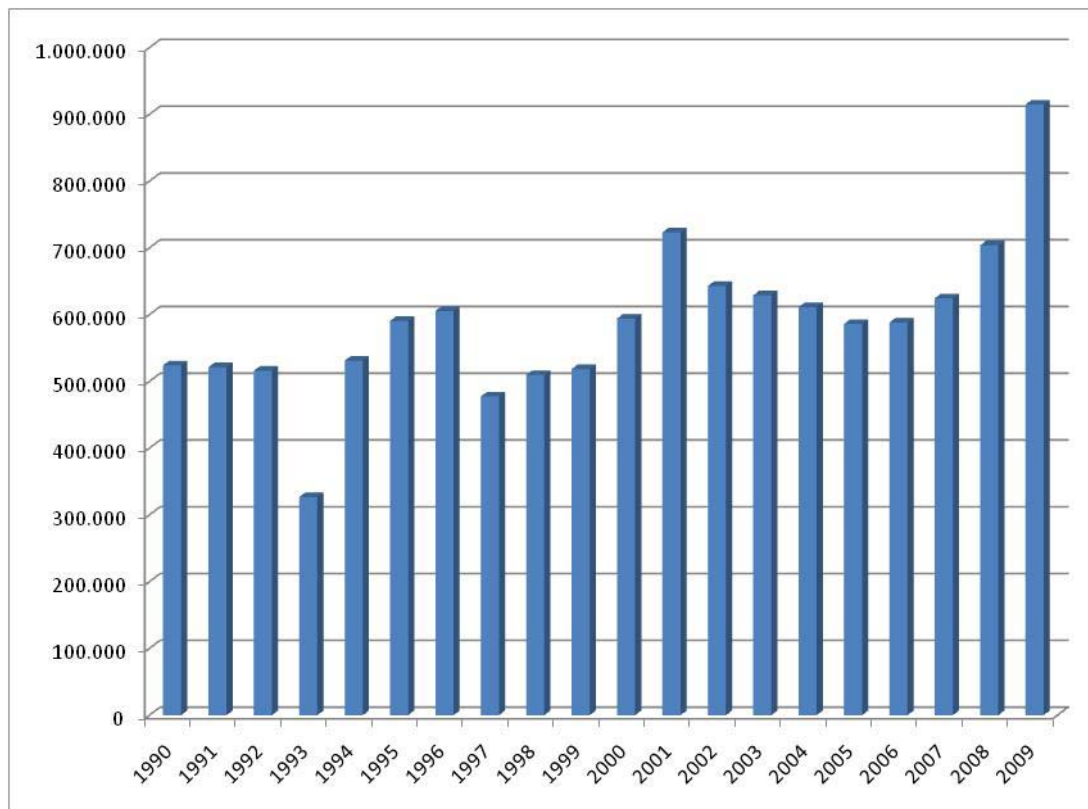
Wie bereits erwähnt, überlagert die Senkung des Körperschaftsteuertarifs 1994 andere Effekte wie die Erhöhung der Hebesätze bei der Grundsteuer und den Gewerbesteuer. Diese Senkung des Körperschaftsteuertarifs wird jedoch 1995 durch die Wiedereinführung des Solidaritätszuschlags konterkariert und zum Teil wieder ausgeglichen. 1996 gibt es außer den Anpassungen der grund- und gewerbesteuerlichen Hebesätze keine Unterschiede. 1997 wird die Vermögensteuer ausgesetzt und 1998 entfällt dann die Gewerbesteuer vom Kapital und es erfolgt eine Absenkung des Solidaritätszuschlags auf 5,5 %.

Die oben angesprochene Reduzierung des Körperschaftsteuertarifs auf 40 % dominiert 1999 und 2000. Die nachfolgende Verminderung auf 25 % wird durch die gleichzeitig eingeführten restriktiveren Abschreibungsmöglichkeiten zum Teil wieder abgemildert, so dass die Steuerlast nicht so stark absinkt, wie dies die Tarifanpassung suggeriert. 2002 wirkt sich die Änderung bei der betrieblichen Altersvorsorge nicht signifikant aus. 2003 wird der Körperschaftsteuersatz temporär auf 26,5 % angehoben, was die leicht erhöhte Belastung erklärt. 2004 wirken sich die Verschärfungen bei der Verlustverrechnung im aktuellen Szenario nicht aus. Im folgenden Jahr gibt es keine relevanten Änderungen.

Die Erhöhung der degressiven Abschreibung im Jahr 2006 wirkt sich auch kaum auf das Ergebnis aus. Erst 2008 gibt es durch die weitere Absenkung des Körperschaftsteuertarifs auf 15 % wieder eine deutliche Veränderung der Steuerbelastung. Die Gegenfinanzierungsmaßnahmen haben nur geringe Effekte (Abschaffung der degressiven Abschreibung) oder greifen im aktuellen Szenario, wie z. B. die Zinsschranke,

nicht, da der Zinsaufwand des Unternehmens über alle Berechnungsperioden kleiner als eine Million ist. Dies ist auch 2009 sehr gut zu beobachten, da die Wiedereinführung der degressiven Abschreibung und die auf drei Millionen erhöhte Freigrenze bei der Zinsschranke kaum Wirkung entfalten.

*Abbildung 93: Steuerbelastung in Deutschland auf Anteilseignerebene im Zeitablauf*



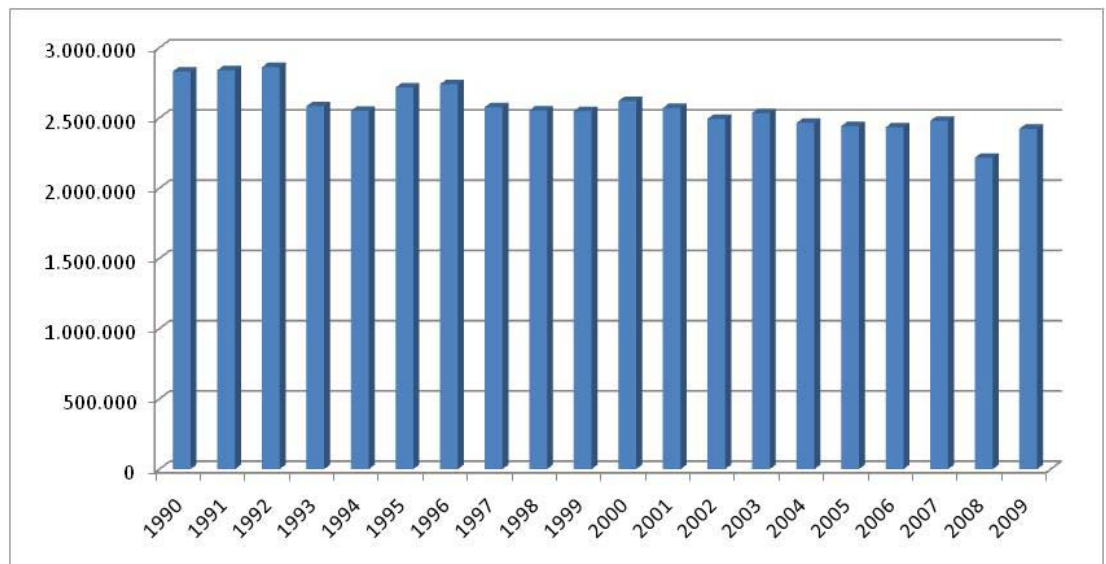
Durch Entlastungen auf der Unternehmensebene, die im Ausschüttungsfall zu höheren steuerpflichtigen Einkünften führen, kann die Einkommenssteuerbelastung der Anteilseigner steigen. Die erhöhten Einkünfte können dann die Effekte, z. B. aus Tarifreduzierungen, kompensieren oder überlagern. Eine isolierte Betrachtung ist somit nur bedingt sinnvoll ist.

Diesen Effekt kann man sehr gut in 2009 beobachten. Dieses Jahr weist mit Abstand die höchste Belastung bei Anteilseignern auf, obwohl der Einkommensteuertarif und der Tarif der Abgeltungsteuer mit am vorteilhaftesten in diesem Jahr sind.

1993 tritt ein seltsamer Effekt auf. Die ESt-Zahlungen der Anteilseigner 3-10 sind durch den erhöhten Sparerfreibetrag niedriger als die Körperschaftsteuerrückstattung und somit entsteht ein Steuerguthaben, das die Belastung mindert.

Ansonsten ist nur der Vergleich zwischen Jahren sinnvoll, die auf der Ebene der Gesellschaft nahezu identische Resultate liefern. Im Jahr 2005 kann man den reduzierten Einkommensteuertarif (der Spitzensteuersatz sank von 45 % auf 42 %) im Vergleich zu 2004 erkennen.

*Abbildung 94: Steuerbelastung in Deutschland auf Gesamtebene im Zeitablauf*



Aufschlussreicher ist dagegen die Gesamtebene. Hierbei ist zu erkennen, dass die Gesamtbelastung nur leicht fällt. Somit wurden Entlastungen auf der Ebene der Körperschaft meistens durch Belastungen auf der Ebene der Besitzer wieder kompensiert. Als Beispiel sei hier der Übergang vom Anrechnungsverfahren auf das Halbeinkünfteverfahren (2001) genannt, welches durch die Abgeltungsteuer (2009) ersetzt wurde.

## 6.9 Recherche und Kontrolle im Webinterface

### 6.9.1 Länderdetails

Mithilfe einer Baumstruktur können, ähnlich wie in der Desktopanwendung (vgl. Abbildung 69), alle Details eines Landes aufgerufen werden.

Abbildung 95: Recherche mithilfe der Baumstruktur

The screenshot shows the Vitax ZEW web application interface. On the left is a navigation tree with categories like 'Germany', '2001\_CURRENT\_LAW', 'PDFs', 'Steuerdaten', and 'Anteilseigner'. The main content area on the right displays details for 'Germany\_2001\_CURRENT\_LAW', including attributes like Land, Jahr, Typ, Wechselkurs, and Währungssymbol. Below this is a 'Vorratsvermögen' section with checkboxes for Last-In-First-Out, First-In-First-Out, and Highest-In-First-Out. At the bottom is a 'Historie' table showing a log of updates.

Historie				
Nutzer	Qualität	Datum & Zeit	Herkunftsland	Kommentar
Unknown	COPIED	26.12.09 16:31	Deutschland_2001_CURRENT_LAW	
mgw	MODIFIED	30.12.09 16:44		
mgw	MODIFIED	19.03.10 15:17		
mgw	MODIFIED	25.03.10 14:04		SolZ-Tarif angepasst an andere Jahre.
mgw	MODIFIED	01.04.10 10:59		Ersetzte WG entfernt.

In Abbildung 95 ist auf der linken Seite der teilweise expandierte Navigationsbaum für die Länderdetails sichtbar. Rechts sind die Details des aktuell ausgewählten Knotens zu sehen. Neben den Länderattributen sieht man auch die gewählte Verbrauchsfolge und die Versionshistorie.<sup>267</sup>

### 6.9.2 Erstellen und Anzeige von Views

Für den Administrator besteht zudem die Möglichkeit, direkt Structured Query Language (SQL) Abfragen (vgl. Abbildung 96) zu hinterlegen. Diese können dann von jedem Benutzer abgerufen und ausgeführt werden. Dadurch ist es auf einfache Art und Weise möglich, komplexe Daten aufzubereiten und somit eine spezielle Sicht (englisch, SQL: View) auf die Datenbank zu erzeugen.

<sup>267</sup> Durch die temporäre Speicherung ging die ursprüngliche Versionshistorie verloren, daher startet diese mit dem Eintrag kopiert von 2001 und ist recht kurz. Dieses Problem wurde inzwischen behoben.

Abbildung 96: Menü zur Auswahl und Eingabe von SQL-Abfragen

**Abschreibungen**

Wirtschaftsgut	Abschreibungsmethode	Param	
Maschine 3	Straight Line	3	Details
Maschine 4	UniversalMethode		Details
Maschine 5	DecliningBalance (German -- schitch to linear.)		Details
Patent	Straight Line	4	Details
Unbebautes Grundstück 1	DecliningBalance (German -- schitch to linear.)		Details

**Abschreibungsmethode für Maschine 2**

Abschreibungsmethode: DecliningBalance (German -- schitch to linear.)  
 Kommentar:  
 Speichername: unbenannt  
 Abschreibbar von AHK [%]: 100.0  
 Auflösung stille Reserven in Jahren: 0

	1	2	3	4	5	6	7	8	9	10
Daten:	25.0	25.0	33.33333333333333	50.0	100.0	0.0	0.0	0.0	0.0	0.0

	1	2	3	4	5	6	7	8	9	10
Absolut:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Relativ:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

	1	2	3	4	5	6	7	8	9	10
AHK:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
RBW:	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Unten ist eine Abfrage für die deutschen Grundsteuertarife exemplarisch aufgeführt. Diese kann im Webinterface unverändert hinterlegt werden (vgl. Abbildung 97). Zur Speicherung und Anzeige muss zudem noch ein eindeutiger Name vergeben und die Spaltennamen definiert werden. Deren Anzahl muss mit den ausgewählten Attributen in der *Select*-Anweisung übereinstimmen.

```
Select distinct l.year, tl.funktion from countrymodel as l,
unternehmensmodell as um, unternehmensmodellvst as umvst,
steuerdaten as sd, steuerobjekt as so, taxbasesammelbaustein as
tb, universaltarif as ut, taxratelevel as tl, bmg_baustein as bmg
where l.unternehmensmodell_id = um.id
and um.nachsteuerum_id = umvst.id
and umvst.steuerdatenunt_id = sd.id
and so.steuerdate_id = sd.id
and so.collectioncomponent_id = tb.id
and tb.tarifimp_id = ut.id
and ut.id = tl.universaltarif_id
and l.type like 'C%'
and l.name like 'GER%'
and ut.tarifoftax = true
and so.name like 'Grund%'
order by l.year
```

Mit „*Select*“ werden die gewünschten Spalten einzelner Tabellen („*from*“) ausgewählt. „*Distinct*“ eliminiert doppelte Ergebnisse. Zur kürzeren Schreibweise werden

Abkürzungen („as“) für die Relationennamen definiert. In der „where“-Bedingung werden zunächst die einzelnen Tabellen miteinander verknüpft.

**Abbildung 97: Eingabe einer SQL-Abfrage**

The screenshot shows the Vitax database interface. At the top, there are logos for Vitax and ZEW (Zentrum für Europäische Wirtschaftsforschung/Center for European Economic Research). Below the logos, there is a navigation bar with the text "Länder | Eigene Steueranzeige | Suchen". On the right side of the navigation bar, there is a button labeled "Öffne DB-Schema".

The main form has the following fields and controls:

- Name:** A text input field containing "Vermögenssteuer DeutschaIn d UE".
- Spaltennamen:** A table with two columns: "Jahr" and "Tarif". There are buttons "Hinzufügen" and "Entfernen" next to the "Tarif" column.
- SQL - Abfrage:** A text area containing the following SQL query:
 

```
Select distinct l.year,
tl.funktion
from countrymodell as l, unternehmensmodell as um, unternehmensmodellvst as umvst,
steuerdaten as sd, steuerobjekt as so, taxbasesammelbaustein as tb, universaltarif as ut,
taxratelevel as tl, bmg_baustein as bmg
where l.unternehmensmodell_id = um.id
and um.nachsteuerum_id = umvst.id
and umvst.steuerdatenunt_id = sd.id
```
- Kommentar:** A large text area for comments.
- At the bottom, there are two buttons: "Vorschau" and "Speichern".

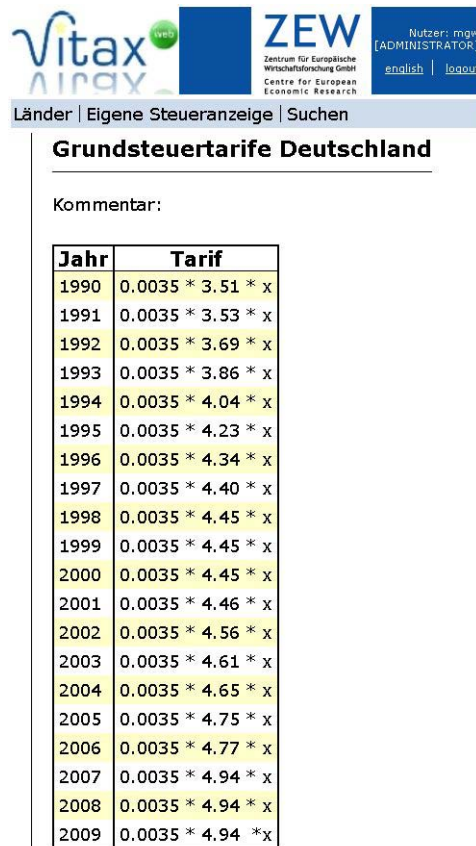
Das Land verweist auf ein Unternehmensmodell, das wiederum auf den steuerspezifischen Teil verweist. Dieser enthält dann eine Verknüpfung zu den Steuerdaten und diese wiederum zu den konkreten Steuerobjekten. Diese enthalten einen Sammelbaustein, der wiederum einen Tarif beherbergt.

Danach wird die Suchanfrage noch auf Deutschland („l.name like 'GER%'“), den aktuellen Rechtsstand („l.type like 'C%'“) und die Grundsteuer („so.name like 'Grund%'“) eingeschränkt. Wobei 'GER%' die Abkürzung für *Germany* und 'C%' für *Current-Law* darstellt. Ebenso wird die Grundsteuer mit 'Grund%' abgekürzt.

Mit „ut.tarifoftax = true“ wird überprüft, ob der Tarif als Tarif dieser Steuer markiert wurde oder nicht. Dadurch werden Tarife von Hilfsrechnungen eliminiert. Allerdings werden Tarife, die nicht entsprechend kenntlich gemacht wurden, dann auch nicht mehr gefunden. Das Ergebnis dieser Abfrage ist in Abbildung 98 zu sehen.

Durch die ausführliche Schreibweise des Tarifs ist es einfach, zwischen der Messzahl und dem Hebesatz zu unterscheiden. Der in dieser Untersuchung herangezogene Hebesatz, für Gemeinden mit mehr als 50.000 Einwohnern, für die Grundsteuer ist von 1990 bis 2009 kontinuierlich von 351 % auf 494 % gestiegen.

Abbildung 98: Ausgabe der Grundsteuertarife im Webinterface



Jahr	Tarif
1990	0.0035 * 3.51 * x
1991	0.0035 * 3.53 * x
1992	0.0035 * 3.69 * x
1993	0.0035 * 3.86 * x
1994	0.0035 * 4.04 * x
1995	0.0035 * 4.23 * x
1996	0.0035 * 4.34 * x
1997	0.0035 * 4.40 * x
1998	0.0035 * 4.45 * x
1999	0.0035 * 4.45 * x
2000	0.0035 * 4.45 * x
2001	0.0035 * 4.46 * x
2002	0.0035 * 4.56 * x
2003	0.0035 * 4.61 * x
2004	0.0035 * 4.65 * x
2005	0.0035 * 4.75 * x
2006	0.0035 * 4.77 * x
2007	0.0035 * 4.94 * x
2008	0.0035 * 4.94 * x
2009	0.0035 * 4.94 * x

Ein Problem ergibt sich durch die rekursive Struktur des Sammelbausteins, die in SQL nicht direkt behandelt werden kann.<sup>268</sup> Zwar kann man bei Kenntnis der konkreten Stufe, auf welcher sich der wirkliche Tarif einer Steuer befindet, diesen auch finden. Es ist jedoch nicht möglich, einfach alle Ebenen zu durchsuchen, bis der tatsächliche Tarif gefunden wurde. In SQL müsste man für jedes Level eine eigene Anfrage schreiben und deren Resultate via „*union*“ miteinander verknüpfen. Allerdings gibt es Datenbank spezifische Erweiterungen von SQL, die die Ermittlung solcher transitiver Hüllen ermöglichen.<sup>269</sup> Alternativ könnte für solche Fälle eine universale Java basierte Lösung angeboten werden.

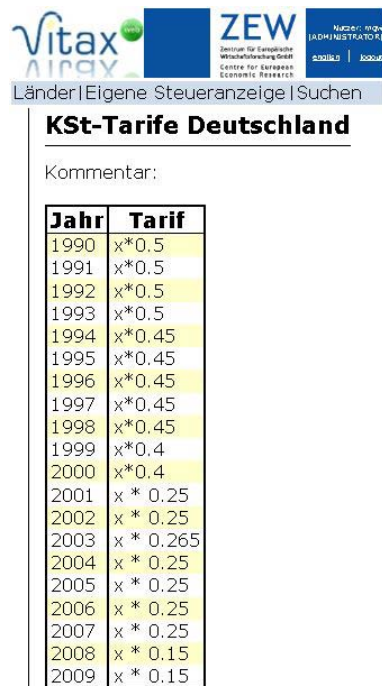
Ähnliche Abfragen wurden benutzt, um die anderen Tarife zu recherchieren. Abbildung 99 zeigt die Entwicklung der Körperschaftsteuertarife.

<sup>268</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 127 ff.

<sup>269</sup> Vgl. Kemper, A./Eickler, A., 2009, S. 127 ff. Seit PostgreSQL 8.4 ist dies auch implementiert.



Abbildung 99: Körperschaftsteuertarife für Deutschland von 1990-2009



**KSt-Tarife Deutschland**

Kommentar:

Jahr	Tarif
1990	$x \cdot 0.5$
1991	$x \cdot 0.5$
1992	$x \cdot 0.5$
1993	$x \cdot 0.5$
1994	$x \cdot 0.45$
1995	$x \cdot 0.45$
1996	$x \cdot 0.45$
1997	$x \cdot 0.45$
1998	$x \cdot 0.45$
1999	$x \cdot 0.4$
2000	$x \cdot 0.4$
2001	$x \cdot 0.25$
2002	$x \cdot 0.25$
2003	$x \cdot 0.265$
2004	$x \cdot 0.25$
2005	$x \cdot 0.25$
2006	$x \cdot 0.25$
2007	$x \cdot 0.25$
2008	$x \cdot 0.15$
2009	$x \cdot 0.15$

Abbildung 100 zeigt die Tarife der Gewerbesteuer vom Ertrag und in Abbildung 101 ist die der Gewerbesteuer vom Kapital zu sehen.

Abbildung 100: Tarif der Gewerbesteuer vom Ertrag



**GewSt vom Ertrag -- Tarife Deutschland**

Kommentar:

Jahr	Tarif
1990	$x \cdot ((5 \cdot 407) / (10000 + 5 \cdot 407))$
1991	$x \cdot ((5 \cdot 382) / (10000 + 5 \cdot 382))$
1992	$x \cdot ((5 \cdot 398) / (10000 + 5 \cdot 398))$
1993	$x \cdot ((5 \cdot 403) / (10000 + 5 \cdot 403))$
1994	$x \cdot ((5 \cdot 406) / (10000 + 5 \cdot 406))$
1995	$x \cdot ((5 \cdot 409) / (10000 + 5 \cdot 409))$
1996	$x \cdot ((5 \cdot 416) / (10000 + 5 \cdot 416))$
1997	$x \cdot ((5 \cdot 424) / (10000 + 5 \cdot 424))$
1998	$x \cdot ((5 \cdot 426) / (10000 + 5 \cdot 426))$
1999	$x \cdot ((5 \cdot 428) / (10000 + 5 \cdot 428))$
2000	$x \cdot ((5 \cdot 428) / (10000 + 5 \cdot 428))$
2001	$x \cdot ((5 \cdot 428) / (10000 + 5 \cdot 428))$
2002	$x \cdot ((5 \cdot 430) / (10000 + 5 \cdot 430))$
2003	$x \cdot ((5 \cdot 431) / (10000 + 5 \cdot 431))$
2004	$x \cdot ((5 \cdot 432) / (10000 + 5 \cdot 432))$
2005	$x \cdot ((5 \cdot 433) / (10000 + 5 \cdot 433))$
2006	$x \cdot ((5 \cdot 433) / (10000 + 5 \cdot 433))$
2007	$x \cdot ((5 \cdot 432) / (10000 + 5 \cdot 432))$
2008	$x \cdot 0.035 \cdot 4.32$
2009	$x \cdot 0.035 \cdot 4.32$



Neben der unterschiedlichen Länge der Zeitreihen, welche durch die Abschaffung der Gewerbesteuer vom Kapital 1998 entsteht, fällt auch der identische Hebesatz ins Auge. Zudem fehlt der Freibetrag in Höhe von 120.000 DM bei der ertragsabhängigen Steuer. 2008 wurde dann die Messzahl geändert.

**Abbildung 101: Tarife der Gewerbesteuer vom Kapital**




Nutzen: mge  
 [ADMINISTRATOR]  
[anmelden](#) | [logout](#)

Länder | Eigene Steueranzeige | Suchen

**GewSt vom Kapital -- Tarife Deutschland**

Kommentar:

Jahr	Untere Grenze	Obere Grenze	Tariffunktion
1990	0	120000	0
1990	120000	9223372036854775807	$((x-120000) * (0.2*407))/10000$
1991	0	120000	0
1991	120000	9223372036854775807	$((x-120000) * (0.2*382))/10000$
1992	0	120000	0
1992	120000	9223372036854775807	$((x-120000) * (0.2*398))/10000$
1993	0	120000	0
1993	120000	9223372036854775807	$((x-120000) * (0.2*403))/10000$
1994	0	120000	0
1994	120000	9223372036854775807	$((x-120000) * (0.2*406))/10000$
1995	0	120000	0
1995	120000	9223372036854775807	$((x-120000) * (0.2*409))/10000$
1996	0	120000	0
1996	120000	9223372036854775807	$((x-120000) * (0.2*416))/10000$
1997	0	120000	0
1997	120000	9223372036854775807	$((x-120000) * (0.2*424))/10000$

Aufgrund der vielen Tarifstufen der Einkommensteuer wurden nur die Jahre selektiert, in denen sich der Tarif geändert hat. Zudem wurde 1990 als Ausgangsbasis gewählt.

Abbildung 102: Zeitreihe der Einkommensteuertarife (nur Änderungsjahre)

Länder   Eigene Steueranzeige   Suchen			
Est-Tarife Deutschland im Jahr der Änderung			
Kommentar:			
Jahr	Untere Grenze	Obere Grenze	Tarif
1990	-9223372036854775808	5617	0
1990	5617	8154	$0.19 * x - 1067$
1990	8154	120042	$(151.94 * ((x - 8100) / 10000) + 1900) * ((x - 8100) / 10000) + 472$
1990	120042	9223372036854775807	$0.53 * x - 22842$
1996	0	12096	0
1996	12096	55728	$(86.63 * ((x - 12042) / 10000) + 2590) * ((x - 12042) / 10000)$
1996	55728	120042	$(151.91 * ((x - 55674) / 10000) + 3346) * ((x - 55674) / 10000) + 12949$
1996	120042	9223372036854775807	$0.53 * x - 22842$
1998	0	12366	0
1998	12366	58644	$(91.19 * ((x - 12312) / 10000) + 2590) * ((x - 12312) / 10000)$
1998	58644	120042	$(151.96 * ((x - 58590) / 10000) + 3434) * ((x - 58590) / 10000) + 13938$
1998	120042	9223372036854775807	$0.53 * x - 22843$
1999	0	13068	0
1999	13068	17064	$(350.55 * ((x - 13014) / 10000) + 2390) * ((x - 13014) / 10000)$
1999	17064	66366	$(101.31 * ((x - 17010) / 10000) + 2670) * ((x - 17010) / 10000) + 1011$
1999	66366	120042	$(151.93 * ((x - 66312) / 10000) + 3669) * ((x - 66312) / 10000) + 16637$
1999	120042	9223372036854775807	$0.53 * x - 22886$
2000	0	13500	0
2000	13500	17496	$(262.76 * ((x - 13446) / 10000) + 2290) * ((x - 13446) / 10000)$
2000	17496	114696	$(113.74 * ((x - 17442) / 10000) + 2500) * ((x - 17442) / 10000) + 957$
2000	114696	9223372036854775807	$0.51 * x - 20576$
2001	0	14094	0
2001	14094	18090	$(387.89 * ((x - 14040) / 10000) + 1990) * ((x - 14040) / 10000)$
2001	18090	107568	$(142.49 * ((x - 18036) / 10000) + 2300) * ((x - 18036) / 10000) + 432$
2001	107568	9223372036854775807	$0.485 * x - 19.299$
2002	0	7236	0
2002	7236	9252	$(786.85 * ((x - 7200) / 10000) + 1990) * ((x - 7200) / 10000)$
2002	9252	55008	$(278.65 * ((x - 9216) / 10000) + 2300) * ((x - 9216) / 10000) + 432$
2002	55008	9223372036854775807	$0.485 * x - 9872$
2004	0	7665	0
2004	7665	12740	$(793.10 * ((x - 7664) / 10000) + 1600) * ((x - 7664) / 10000)$
2004	12740	52152	$(265.78 * ((x - 12739) / 10000) + 2405) * ((x - 12739) / 10000) + 1016$
2004	52152	9223372036854775807	$0.45 * x - 8845$
2005	0	7665	0
2005	7665	12740	$(883.74 * ((x - 7664) / 10000) + 1500) * ((x - 7664) / 10000)$
2005	12740	52152	$(228.74 * ((x - 12739) / 10000) + 2397) * ((x - 12739) / 10000) + 989$
2005	52152	9223372036854775807	$0.42 * x - 7914$
2007	-9223372036854775808	7665	0
2007	7665	12740	$(883.7 * ((x - 7664) / 10000) + 1500) * ((x - 7664) / 10000)$
2007	12740	52152	$(228.74 * ((x - 12739) / 10000) + 2397) * ((x - 12739) / 10000) + 989$
2007	52152	250001	$0.42 * x - 7914$
2007	250001	9223372036854775807	$x * 0.45 - 15414$
2009	0	7835	0
2009	7835	13140	$(939.68 * ((x - 7834) / 10000) + 1400) * ((x - 7834) / 10000)$
2009	13140	52552	$(228.74 * ((x - 13139) / 10000) + 2397) * ((x - 13139) / 10000) + 1007$
2009	52552	250401	$0.42 * x - 8064$
2009	250401	9223372036854775807	$x * 0.45 - 15576$

Beim Vermögensteuertarif fällt die *if*-Abfrage ins Auge. Diese implementiert eine Freigrenze, in der die Steuer für alle Werte kleiner oder gleich 20.000 DM null wird. Darüber wird die komplette Bemessungsgrundlage mit 0.6 % multipliziert. Alternativ könnte dies auch durch einen nicht additiven Stufentarif abgebildet werden. Dies würde dann ähnlich wie bei der Gewerbesteuer vom Ertrag (vgl. Abbildung 100) aussehen, jedoch ohne die Subtraktion der unteren Stufengrenze. Diese Art der Modellierung würde eher der erwarteten Darstellung entsprechen. Die obige Formel wurde hier gewählt, um auch andere Möglichkeiten aufzuzeigen und die Flexibilität des Universaltarifs zu demonstrieren.

Abbildung 103: Tarife der Vermögensteuer auf Unternehmensebene



Jahr	Tarif
1990	$\text{if}(x \leq 20000, 0, x * 0.006)$
1991	$\text{if}(x \leq 20000, 0, x * 0.006)$
1992	$\text{if}(x \leq 20000, 0, x * 0.006)$
1993	$\text{if}(x \leq 20000, 0, x * 0.006)$
1994	$\text{if}(x \leq 20000, 0, x * 0.006)$
1995	$\text{if}(x \leq 20000, 0, x * 0.006)$
1996	$\text{if}(x \leq 20000, 0, x * 0.006)$

Bevor die Erhebung der Vermögensteuer im Jahre 1997 ausgesetzt wurde, wurde der Steuersatz nur einmal seit 1990 und zwar im Jahre 1995 für die Anteilseigner erhöht. Das weiterhin begünstigte Betriebsvermögen wird in einem anderen Baustein separat veranlagt.

Abbildung 104: Vermögensteuertarife für Anteilseigner



Jahr	Obere Grenze	Untere Grenze	Funktion
1990	0	140000	0
1990	140000	9223372036854775807	$(x - 140000) * 0.005$
1991	0	140000	0
1991	140000	9223372036854775807	$(x - 140000) * 0.005$
1992	0	140000	0
1992	140000	9223372036854775807	$(x - 140000) * 0.005$
1993	0	140000	0
1993	140000	9223372036854775807	$(x - 140000) * 0.005$
1994	0	140000	0
1994	140000	9223372036854775807	$(x - 140000) * 0.005$
1995	0	140000	0
1995	140000	9223372036854775807	$(x - 140000) * 0.01$
1996	0	140000	0
1996	140000	9223372036854775807	$(x - 140000) * 0.01$

Abgesehen von Freigrenzen<sup>270</sup> und evtl. vorhandenen Gleitregelungen<sup>271</sup> entsprechen die Tarife des Unternehmens beim Solidaritätszuschlag denen auf Anteilseignerebene

<sup>270</sup> §3 Abs. 3 Nr. 1 und 2 SolZG.

<sup>271</sup> §4 SolZG.

ne. Daher wurde auf eine explizite Darstellung verzichtet. Für die Jahre 1991 und 1992 wurde der Zuschlag nur ein halbes Jahr lang erhoben, sodass in Abbildung 105 nur der halbe Tarifsatz (vgl. 6.5) zu sehen ist.

*Abbildung 105: Tarifsätze für den Solidaritätszuschlag*



**SolZ Deutschland UE**

Kommentar:

Jahr	Tarif
1991	x * 0.0375
1992	x * 0.0375
1995	x * 0.075
1996	x * 0.075
1997	x * 0.075
1998	x * 0.055
1999	x * 0.055
2000	x * 0.055
2001	x * 0.055
2002	x * 0.055
2004	x * 0.055
2005	x * 0.055
2006	x * 0.055
2007	x * 0.055
2008	x * 0.055
2009	x * 0.055

### 6.9.3 Kontrolle

Neben der Möglichkeit, sich entsprechend der Baumstruktur einzelne Implementierungsdetails gezielt anzuschauen, besteht auch die Möglichkeit, sich fast alle steuerlichen Parameter eines Landes kompakt in einer druckbaren Übersicht (vgl. Abbildung 95 - Button auf der rechten Seite) ausgeben zu lassen. Dies soll den Abgleich zwischen recherchierten und eingegebenen Daten erleichtern und somit die Fehlerwahrscheinlichkeit reduzieren.

In Abbildung 106 ist ein Ausschnitt des Resultats dieser Funktion zu sehen. Neben den grundlegenden Informationen zum selektierten Land sind die gewählte Verbrauchsfolge, die Daten zur betrieblichen Altersvorsorge, die gewählten Abschreibungsarten für die einzelnen Wirtschaftsgüter inkl. deren Parameter und exemplarisch die Grundsteuer zu sehen.

Abbildung 106: Ausschnitt der kompakten Länderübersicht

**Germany \_1990\_CURRENT\_LAW**

Land: Germany  
 Jahr: 1990  
 Typ: CURRENT\_LAW  
 Wechselkurs: 1000  
 Währungssymbol: EUR  
 Stille Reserven: ☒  
 Höhere Ausschüttungen: ☐  
 Gewährleistungsrückstellung: ☒  
 Kommentar:  
 Vorratsvermögen - Verfahren: Last-In-First-Out

**Betriebliche Altersvorsorge**

Berücksichtige Eintrittsalter: ☒  
 Berücksichtigtes Eintrittsalter: 30  
 Kalkulationszinssatz (steuerlich) [%]: 60  
 Verfahren: Teilwertverfahren (Deckungsbeitrag)  
 Trendberücksichtigung: ☐  
 Inflation der Rentenzahlungen berücksichtigen: ☐  
 Durchführungsform: Pensionsrückstellung

**Abschreibungen**

Abschreibbares Wirtschaftsgut	Abschreibungsmethode	Param
Betriebsausstattung	Straight Line	3 Jahre
Büro- und Geschäftsgebäude	Straight Line	25 Jahre
Fabrikationsgebäude	Straight Line	25 Jahre
Geschäftsausstattung	Declining Balance (German -- switch to linear.)	6 Jahre; 30.0%
Lizenz	Straight Line	4 Jahre
Maschine 1	Straight Line	3 Jahre
Maschine 2	DecliningBalance (German -- schitch to linear.)	4 Jahre; 30.0%
Maschine 3	DecliningBalance (German -- schitch to linear.)	4 Jahre; 30.0%
Maschine 4	DecliningBalance (German -- schitch to linear.)	5 Jahre; 30.0%
Maschine 5	DecliningBalance (German -- schitch to linear.)	7 Jahre; 30.0%
Patent	Straight Line	4 Jahre
Unbebautes Grundstück 1	Dummy	

**Steuerdaten > Grundsteuer**

Von	Bis	Funktion
0	9223372036854775807	0.0035 * 3.51 * x

**Steuerdaten > Grundsteuer > taxbase**

Name	Funktion	Kommentar
Aktiva	x	
Grundstücke	x	
Büro- und Geschäftsgebäude	ahk *0.2	
Fabrikationsgebäude	ahk *0.2	
Unbebautes Grundstück 1	ahk *0.3	

Um eine noch annehmbare Größe zu gewährleisten, werden nicht bei alle Details angezeigt. Bei den Abschreibungen wird z.B. auf die Darstellung der Universalmethode (vgl. Abbildung 96) verzichtet. Diese sind aber über das Land an sich gut kontrollierbar, sodass nur ein kurzer Überblick gegeben wird, der primär zur Kontrolle der gewählten Methode dient.

#### 6.9.4 Suche nach Steuern und Dokumenten

In den noch recht rudimentär<sup>272</sup> implementierten Suchanfragen können bestimmte Steuerarten für ausgewählte Länder gesucht werden. In Abbildung 107 wird nach

<sup>272</sup> Insbesondere die Länderauswahl ist noch zu verbessern.



einer Steuer vom Typ *INCOME\_TAX* gesucht, deren Name „Eink“ enthält. „%“ dient hierbei als Platzhalter für unbekannte Namensteile. Die Suche soll in den Ländern der Zeitreihe und nur für die Anteilseignerebene erfolgen.

Abbildung 107: Suche nach Steuern

**Steuer:**  
( '\_' dient als Platzhalter für ein Zeichen, '%' für 0 oder mehrere Zeichen)  
%Eink%  
Suche Steuer

**Typ:**  
INCOME\_TAX

In Unternehmenssteuerdaten suchen: ☐  
In AE Steuerdaten suchen: ☒

**Einbezogene Länder:**

Country/Year	Selected
France _2008_CURRENT_LAW	<input type="checkbox"/>
Germany _1990_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1990_OTHER_Test	<input checked="" type="checkbox"/>
Germany _1991_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1992_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1993_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1994_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1995_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1996_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1997_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1998_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _1999_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2000_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2001_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2002_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2003_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2004_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2005_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2006_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2007_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2008_CURRENT_LAW	<input checked="" type="checkbox"/>
Germany _2008_CURRENT_LAW_ENTREPRENEURS_OR_PARTNERSHIP	<input type="checkbox"/>
Germany _2009_CURRENT_LAW	<input checked="" type="checkbox"/>
Italy _2008_CURRENT_LAW	<input type="checkbox"/>
Netherlands Netherlands _2008_CURRENT_LAW	<input type="checkbox"/>
Spain _2008_CURRENT_LAW	<input type="checkbox"/>

Alle abwählen

Unten ist das Resultat der obigen Suchanfrage zu sehen. Die Steuer wurde in jedem Jahr auf der Ebene der Anteilseigner gefunden. Das gewünschte Resultat kann nun nach weiteren Details inspiziert werden. In Abbildung 108 wurde der Einkommensteuertarif für 1990 ausgewählt. Zudem sind die Bemessungsgrundlagen ersichtlich.

Abbildung 108: Resultat der Suche

**zurück**

**Germany**  
1990\_CURRENT\_LAW  
Einkommensteuer  
2 \* EST-Zahlung eine Ehepartner  
\* **EST-Zahlung ein Ehepartner**  
\* EST-Zahlung virtuelle Einkünfte  
\* KSt Anrechnung

**Germany**  
1990\_OTHER\_Test  
Germany \_1991\_CURRENT\_LAW  
Germany \_1992\_CURRENT\_LAW  
Germany \_1993\_CURRENT\_LAW  
Germany \_1994\_CURRENT\_LAW  
Germany \_1995\_CURRENT\_LAW  
Germany \_1996\_CURRENT\_LAW  
Germany \_1997\_CURRENT\_LAW

**EST-Zahlung ein Ehepartner**

Typ: ☐ ☒ ☐  
Kommentar: ☐ ☒ ☐

**Enthaltene Bemessungsgrundlagen**

Operation	Name	Typ
-	Kirchensteuer Abzug	Sammelbaustein
+	BMG EST ein Ehepartner	Andere Steuer

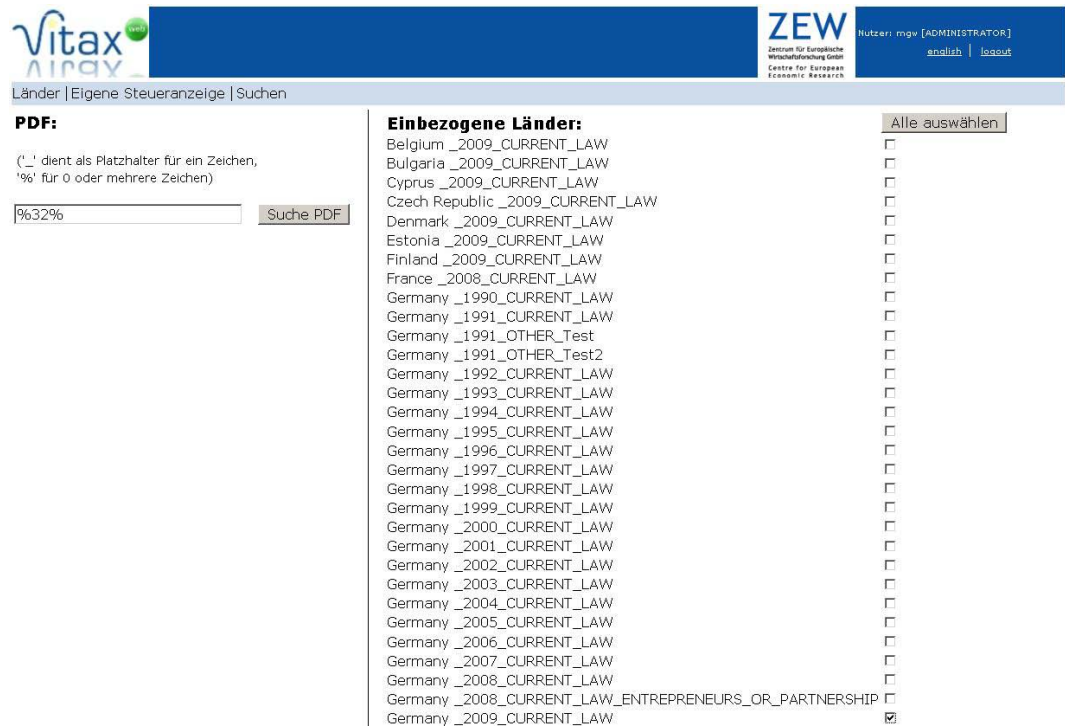
**Universaler Tarif - Est\_Tarif\_D1990**

Von	Bis	Funktion
-9223372036854775808	5617	0
5617	8154	0.19 * x -1067
8154	120042	(151.94*((x-8100)/10000)+1900)*((x-8100)/10000)+472
120042	9223372036854775807	0.53*x-22842

Stufentarif (additive Stufen): ☐  
Inflationierung: ☐  
In sich abzugsfähig: ☐  
Kommentar: ☐

**PDF Dateien**  
Hinzufügen

Abbildung 109: Suchmaske für PDF-Dateien



**Vitax** **ZEW** Zentrum für Europäische Wirtschaftsforschung GmbH  
Centre for European Economic Research Nutzer: mgy [ADMINISTRATOR] [english](#) | [logout](#)

[Länder](#) | [Eigene Steueranzeige](#) | [Suchen](#)

**PDF:**

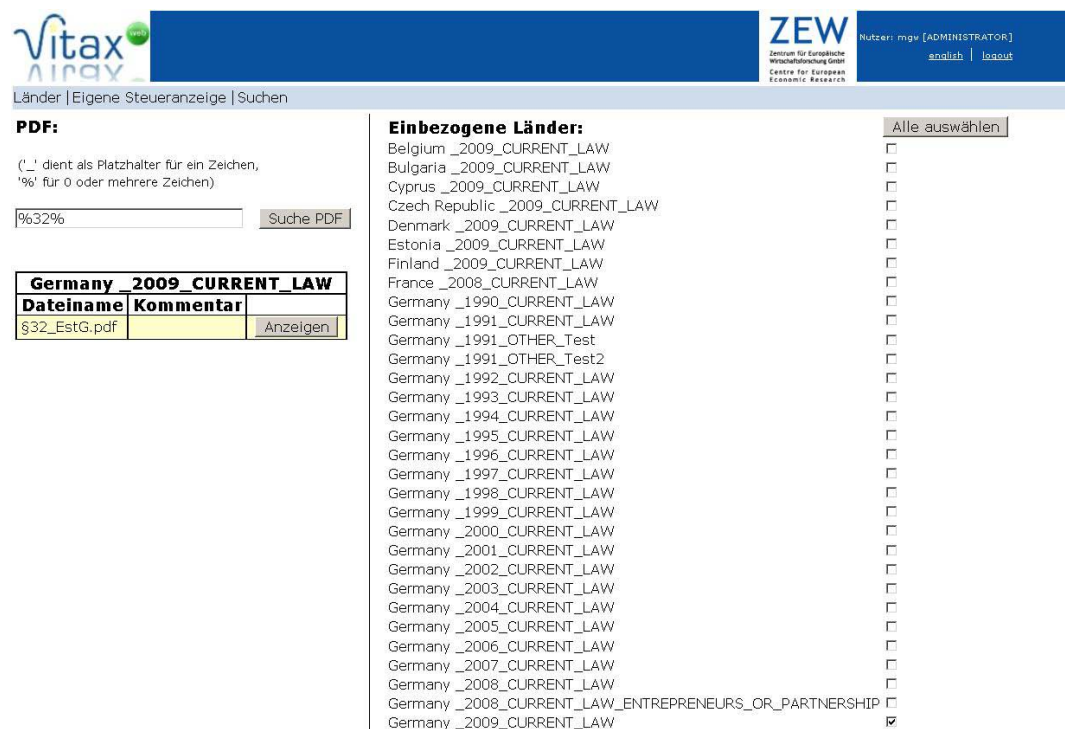
('\_' dient als Platzhalter für ein Zeichen,  
'%' für 0 oder mehrere Zeichen)

**Einbezogene Länder:**

Belgium _2009_CURRENT_LAW	<input type="checkbox"/>
Bulgaria _2009_CURRENT_LAW	<input type="checkbox"/>
Cyprus _2009_CURRENT_LAW	<input type="checkbox"/>
Czech Republic _2009_CURRENT_LAW	<input type="checkbox"/>
Denmark _2009_CURRENT_LAW	<input type="checkbox"/>
Estonia _2009_CURRENT_LAW	<input type="checkbox"/>
Finland _2009_CURRENT_LAW	<input type="checkbox"/>
France _2008_CURRENT_LAW	<input type="checkbox"/>
Germany _1990_CURRENT_LAW	<input type="checkbox"/>
Germany _1991_CURRENT_LAW	<input type="checkbox"/>
Germany _1991_OTHER_Test	<input type="checkbox"/>
Germany _1991_OTHER_Test2	<input type="checkbox"/>
Germany _1992_CURRENT_LAW	<input type="checkbox"/>
Germany _1993_CURRENT_LAW	<input type="checkbox"/>
Germany _1994_CURRENT_LAW	<input type="checkbox"/>
Germany _1995_CURRENT_LAW	<input type="checkbox"/>
Germany _1996_CURRENT_LAW	<input type="checkbox"/>
Germany _1997_CURRENT_LAW	<input type="checkbox"/>
Germany _1998_CURRENT_LAW	<input type="checkbox"/>
Germany _1999_CURRENT_LAW	<input type="checkbox"/>
Germany _2000_CURRENT_LAW	<input type="checkbox"/>
Germany _2001_CURRENT_LAW	<input type="checkbox"/>
Germany _2002_CURRENT_LAW	<input type="checkbox"/>
Germany _2003_CURRENT_LAW	<input type="checkbox"/>
Germany _2004_CURRENT_LAW	<input type="checkbox"/>
Germany _2005_CURRENT_LAW	<input type="checkbox"/>
Germany _2006_CURRENT_LAW	<input type="checkbox"/>
Germany _2007_CURRENT_LAW	<input type="checkbox"/>
Germany _2008_CURRENT_LAW	<input type="checkbox"/>
Germany _2008_CURRENT_LAW_ENTREPRENEURS_OR_PARTNERSHIP	<input type="checkbox"/>
Germany _2009_CURRENT_LAW	<input checked="" type="checkbox"/>

Eine ähnliche Suche ist auch für PDF-Dokumente möglich. Die Suche bezieht sich hier jedoch auf den Namen des Dokumentes. Die Suche nach einer Dokumentation zu § 32 d EStG (Abgeltungsteuertarif) für D 2009 wird in Abbildung 109 gezeigt. Das Resultat (ein Dokument) ist in Abbildung 110 ersichtlich. Die Resultate werden auf der Seite der Suchanfrage eingeblendet.

Abbildung 110: Resultat der PDF-Suche



**Vitax** **ZEW** Zentrum für Europäische Wirtschaftsforschung GmbH  
Centre for European Economic Research Nutzer: mgy [ADMINISTRATOR] [english](#) | [logout](#)

[Länder](#) | [Eigene Steueranzeige](#) | [Suchen](#)

**PDF:**

('\_' dient als Platzhalter für ein Zeichen,  
'%' für 0 oder mehrere Zeichen)

Germany_2009_CURRENT_LAW		
Dateiname	Kommentar	
§32_EstG.pdf	Anzeigen	<input type="button" value="Anzeigen"/>

**Einbezogene Länder:**

Belgium _2009_CURRENT_LAW	<input type="checkbox"/>
Bulgaria _2009_CURRENT_LAW	<input type="checkbox"/>
Cyprus _2009_CURRENT_LAW	<input type="checkbox"/>
Czech Republic _2009_CURRENT_LAW	<input type="checkbox"/>
Denmark _2009_CURRENT_LAW	<input type="checkbox"/>
Estonia _2009_CURRENT_LAW	<input type="checkbox"/>
Finland _2009_CURRENT_LAW	<input type="checkbox"/>
France _2008_CURRENT_LAW	<input type="checkbox"/>
Germany _1990_CURRENT_LAW	<input type="checkbox"/>
Germany _1991_CURRENT_LAW	<input type="checkbox"/>
Germany _1991_OTHER_Test	<input type="checkbox"/>
Germany _1991_OTHER_Test2	<input type="checkbox"/>
Germany _1992_CURRENT_LAW	<input type="checkbox"/>
Germany _1993_CURRENT_LAW	<input type="checkbox"/>
Germany _1994_CURRENT_LAW	<input type="checkbox"/>
Germany _1995_CURRENT_LAW	<input type="checkbox"/>
Germany _1996_CURRENT_LAW	<input type="checkbox"/>
Germany _1997_CURRENT_LAW	<input type="checkbox"/>
Germany _1998_CURRENT_LAW	<input type="checkbox"/>
Germany _1999_CURRENT_LAW	<input type="checkbox"/>
Germany _2000_CURRENT_LAW	<input type="checkbox"/>
Germany _2001_CURRENT_LAW	<input type="checkbox"/>
Germany _2002_CURRENT_LAW	<input type="checkbox"/>
Germany _2003_CURRENT_LAW	<input type="checkbox"/>
Germany _2004_CURRENT_LAW	<input type="checkbox"/>
Germany _2005_CURRENT_LAW	<input type="checkbox"/>
Germany _2006_CURRENT_LAW	<input type="checkbox"/>
Germany _2007_CURRENT_LAW	<input type="checkbox"/>
Germany _2008_CURRENT_LAW	<input type="checkbox"/>
Germany _2008_CURRENT_LAW_ENTREPRENEURS_OR_PARTNERSHIP	<input type="checkbox"/>
Germany _2009_CURRENT_LAW	<input checked="" type="checkbox"/>

## 6.10 Wechsel des ökonomischen Rahmenmodells

Nachdem die Steuerdaten nun vorliegen, ist es relativ einfach, andere Szenarien zu berechnen. Dazu genügt es, die Schritte aus 6.3 zu wiederholen. Ist ein solches Modell jedoch erst einmal erstellt, kann es separat gespeichert<sup>273</sup> und jederzeit einfach geladen werden. Hierdurch wird das vorherige Modell ersetzt.

Im Folgenden wird zunächst eine große Gesellschaft aus der Energiebranche betrachtet. Dieser Industriezweig weist i. d. R. eine höhere Rendite bei gleichzeitig höherer Anlagenintensität auf. Danach wird ein großes Unternehmen des Verarbeitenden Gewerbes untersucht. Zum Schluss wird der ursprüngliche Ausgangsfall durch einen Schock in Periode 6, wie z.B. eine Finanzkrise, modifiziert und analysiert.

### 6.10.1 Branchenwechsel

Die Kennzahlen wurden aus der Amadeus-Datenbank<sup>274</sup> für das Jahr 2001 im Rahmen eines Projektes für die Europäische Kommission extrahiert. Im Gegensatz zu den Bundesbankdaten sind hier Unternehmen aus allen Mitgliedern der Europäischen Union (27) in die Berechnung der Kennzahlen eingeflossen.

*Tabelle 7: Kennzahlen für die Energiebranche und das Verarbeitende Gewerbe (groß)*

Financial Ratio	Energy	Manufacturing
Profit/loss for period	14.038.918	5.087.719
Total assets	507.777.252	158.673.640
Sales	296.484.315	169.088.711
Share of tangible fixed assets	42,85%	33,66%
Return on sales	4,74%	3,01%
Return on Equity	6,60%	8,07%
Equity ratio	41,87%	39,75%
Return on assets	5,50%	5,91%
Inventories to capital	5,10%	19,20%
Costs for personnel to turnover	11,51%	20,93%

*Quelle: Project for the EU Commission TAXUD/2007/DE/325*

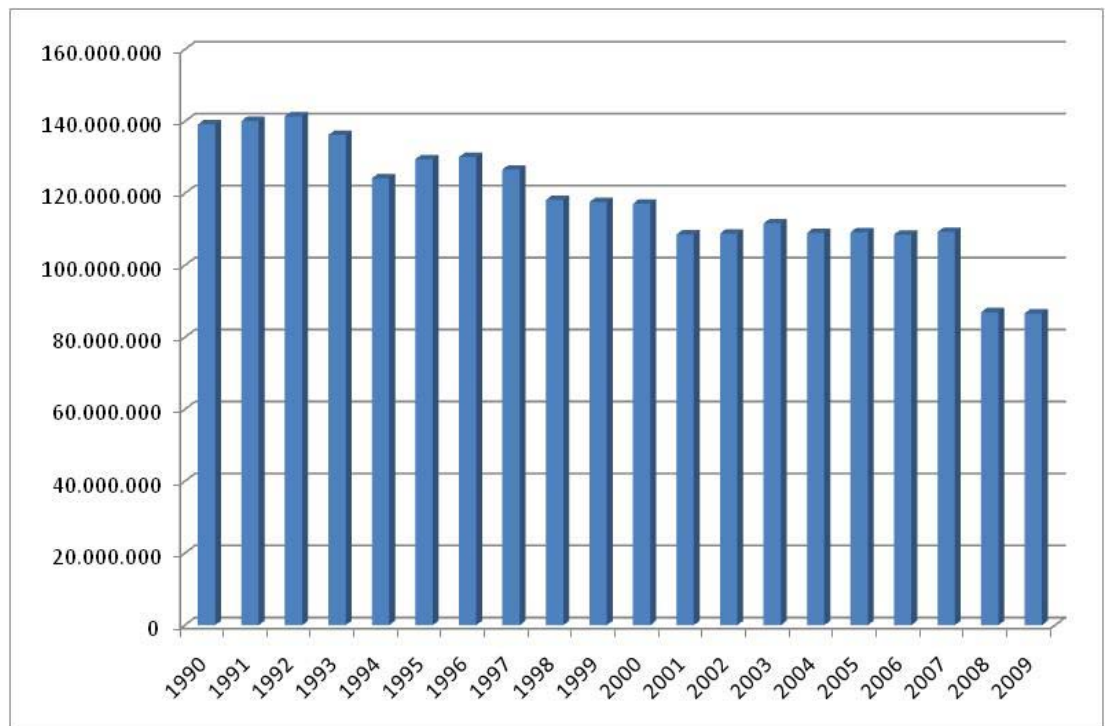
<sup>273</sup> Dies erfolgt außerhalb der Datenbank in XML-Dateien. Das ökonomische Modell ist kein Bestandteil der Datenbank und muss immer separat gespeichert werden. Entweder als eigenständiges Modell (\*.tax), das auch Steuermodellierungen enthalten kann oder als reine Unternehmensdaten (\*.cdt). Aus einem solchen eigenständigen Modell können jederzeit die reinen Unternehmensdaten extrahiert werden.

<sup>274</sup> Siehe <http://www.bvdep.com/de/AMADEUS.html>.



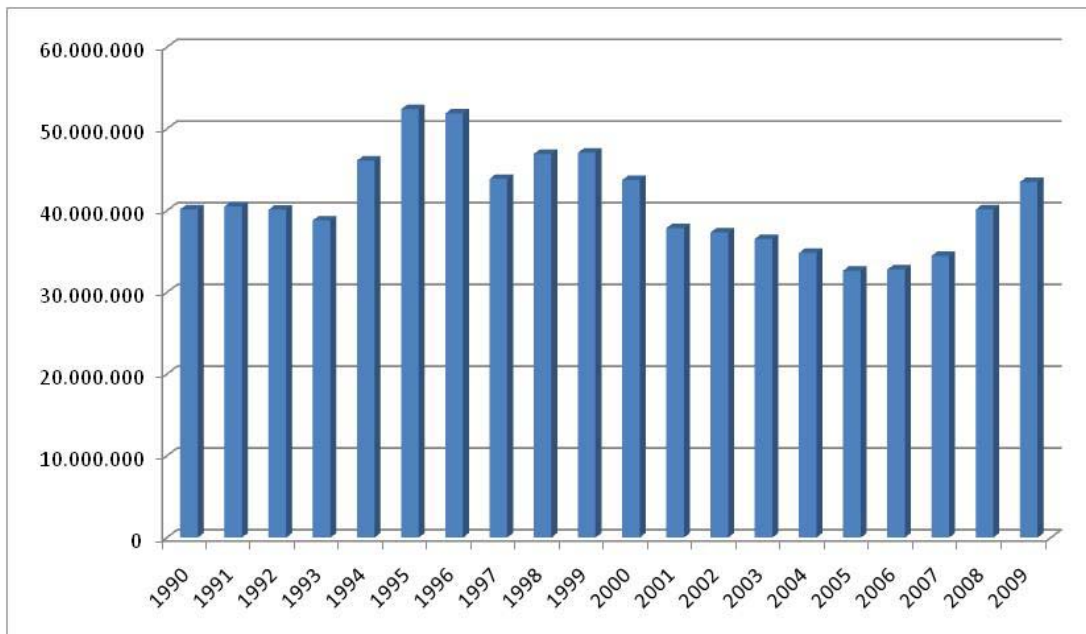
Auf der Ebene der Gesellschaft kann man sehr gut den Tarifverlauf der Körperschaftsteuer ablesen. Die Reduktionen fallen im allgemeinen geringer aus als die Satzänderungen erwarten lassen würden. Dies ist neben den häufig einhergehenden Verschlechterungen der Bemessungsgrundlage auch auf Zinseffekte zurückzuführen. Gesparte Steuerzahlungen machen entweder eine geringere Kreditaufnahme notwendig oder können zum Habenzins angelegt werden und erhöhen somit wieder den zu versteuernden Gewinn.

*Abbildung 111: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Unternehmen)*



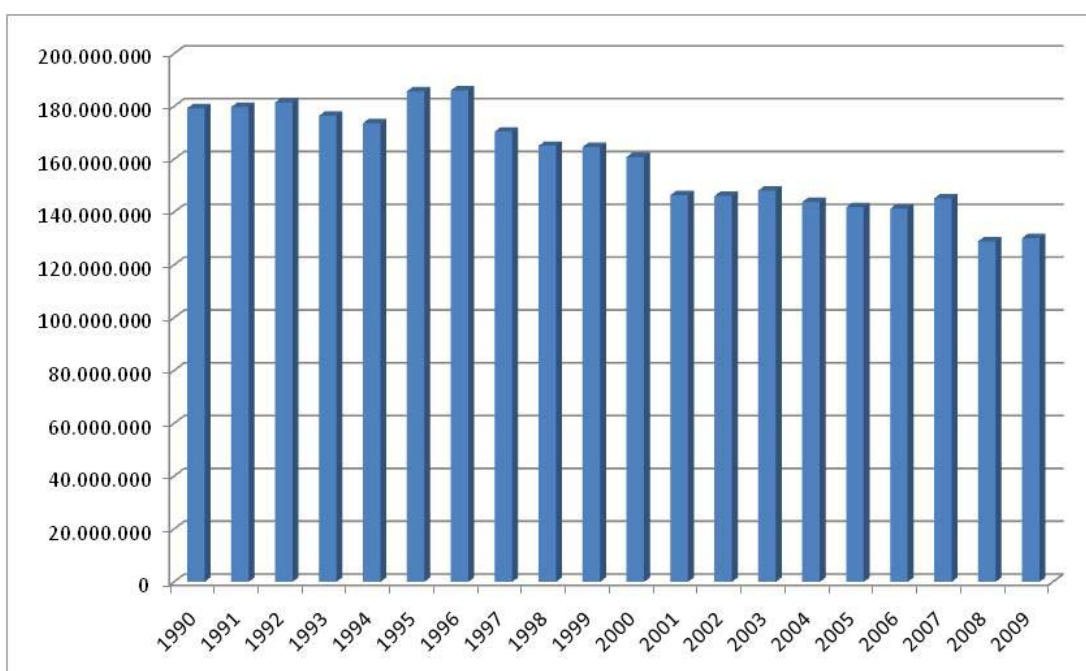
Wie erwartet hat der Wegfall der Substanzsteuern 1997 und 1998, bei einer hoch profitablen Firma, nur geringe Effekte.

**Abbildung 112: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Anteilseigner)**



Für die Anteilseigner spielt der Sparerfreibetrag nur noch eine untergeordnete Rolle. Vielmehr schlagen die Veränderungen bei der Körperschaftsteuer aufgrund erhöhter Ausschüttungen voll durch. 1995 kommt es durch den Solidaritätszuschlag und die erhöhte Vermögensteuer zu einer zusätzlichen Belastung, bevor die Vermögensteuer 1997 ausgesetzt wird. 1998 überwiegt der Effekt der Abschaffung der Gewerbesteuer vom Kapital auf Unternehmensebene im Vergleich zur Reduktion des Solidaritätszuschlages. 2005 macht sich der reduzierte Einkommensteuertarif bemerkbar.

**Abbildung 113: Steuerbelastung in Deutschland im Zeitablauf (Energiebranche - Gesamtebene)**

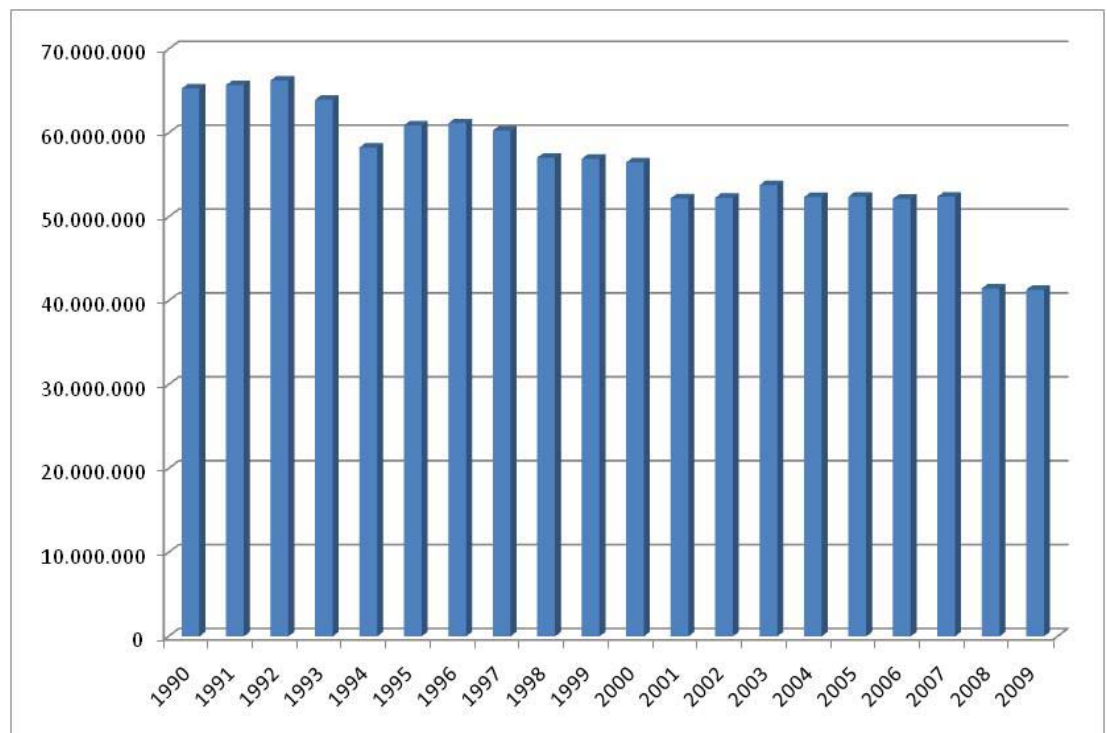


Die Gesamtebene wird ebenso wie bei der Unternehmung von der Körperschaftsteuer dominiert.

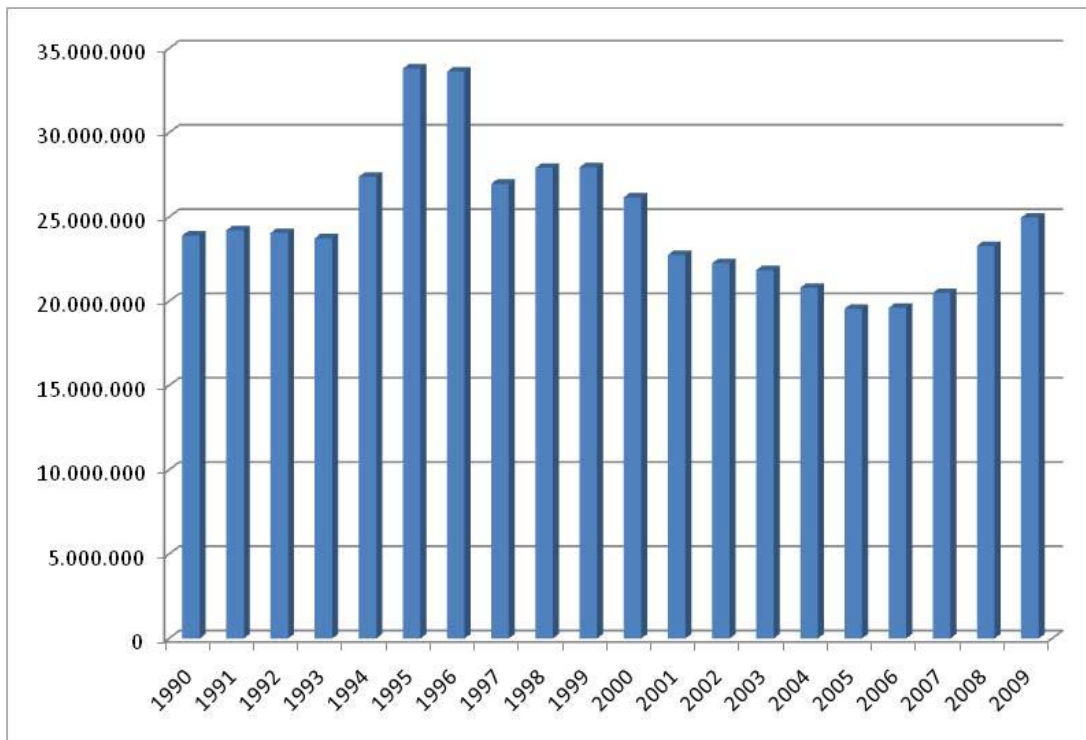
#### 6.10.2 Wechsel der Unternehmensgröße

Beim Wechsel der Unternehmensgröße nähern sich die Effekte denen des Energieunternehmens an. Sie sind jedoch nicht so stark ausgeprägt. Auf eine explizite Beschreibung wird daher verzichtet.

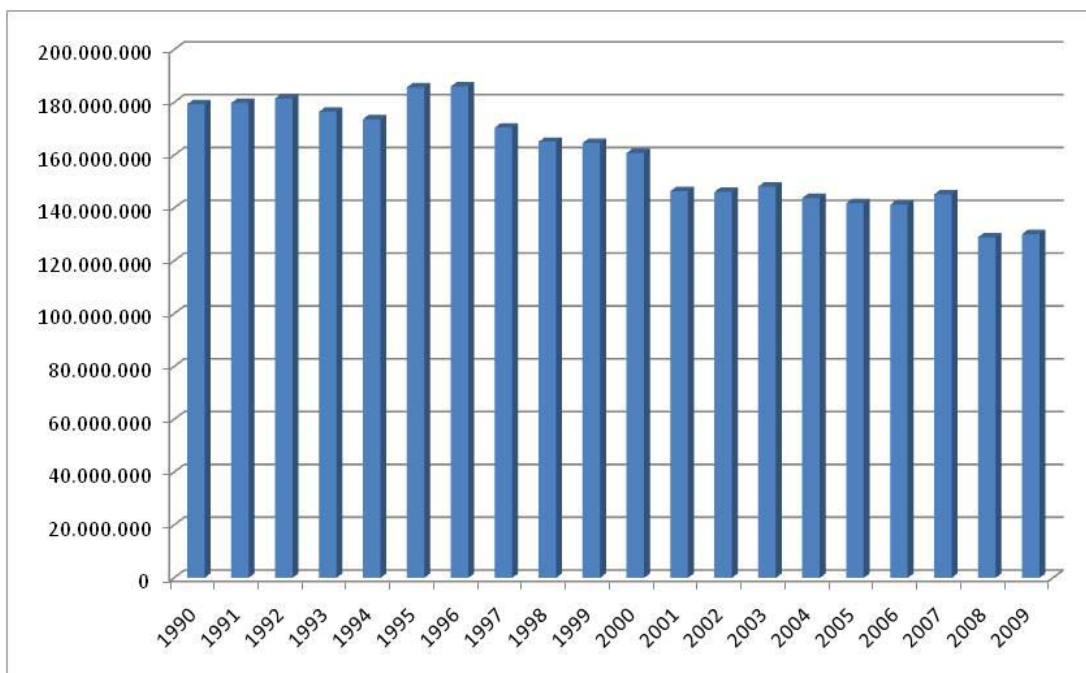
*Abbildung 114: Verarbeitendes Gewerbe groß (Unternehmensebene)*



*Abbildung 115: Verarbeitendes Gewerbe groß (Ebene der Eigner)*



*Abbildung 116: Verarbeitendes Gewerbe groß (Gesamt)*

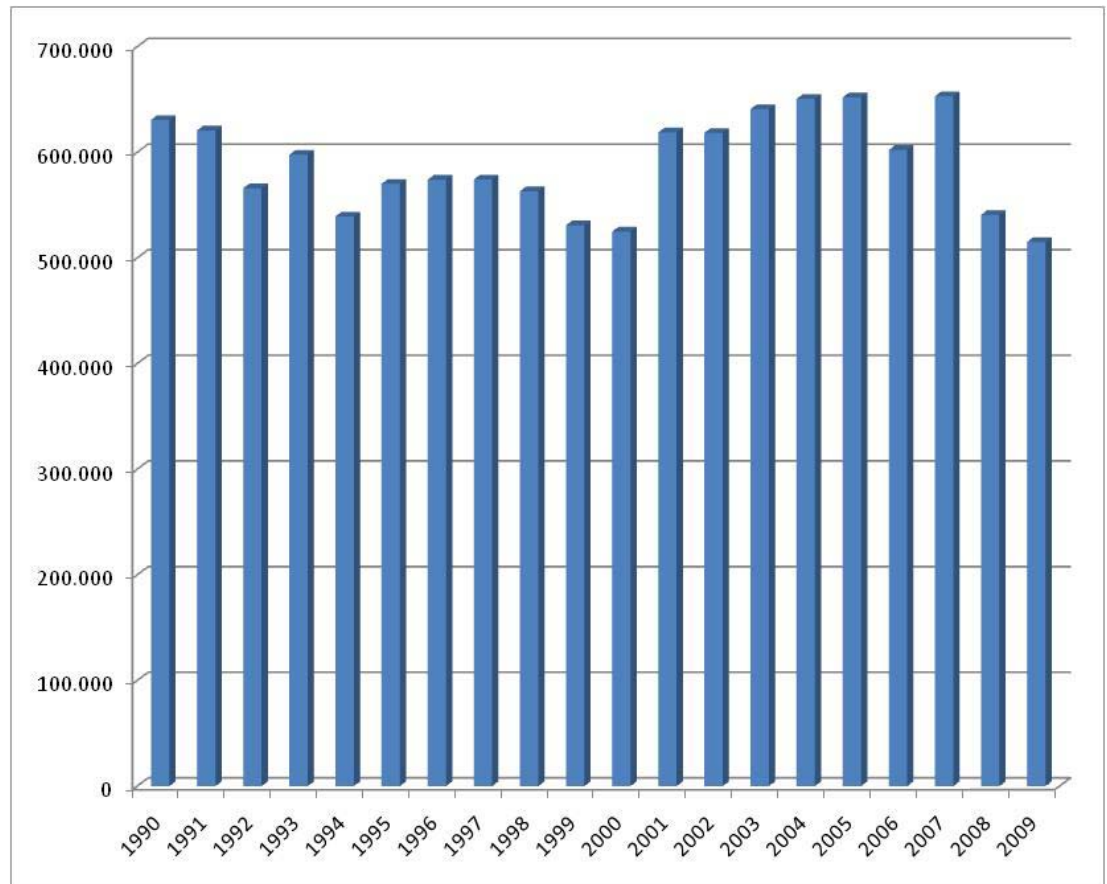


### 6.10.3 Finanzkrise

Die Simulation unterstellt generell gestiegene Zinssätze zur Aufnahme von Fremdkapital. 10 % anstelle von 7 % für kurzfristige und 9 % anstelle von 6 % für langfristige Verbindlichkeiten. Zudem fallen in Periode 6 außerordentliche Aufwendungen

in Höhe von 1.500.000 € an. Eine Anpassung des Produktions- und Absatzplans wurde aus Gründen der besseren Vergleichbarkeit nicht vorgenommen.

*Abbildung 117: Steuerbelastung in Deutschland im Zeitablauf (Finanzkrise - Unternehmensebene)*



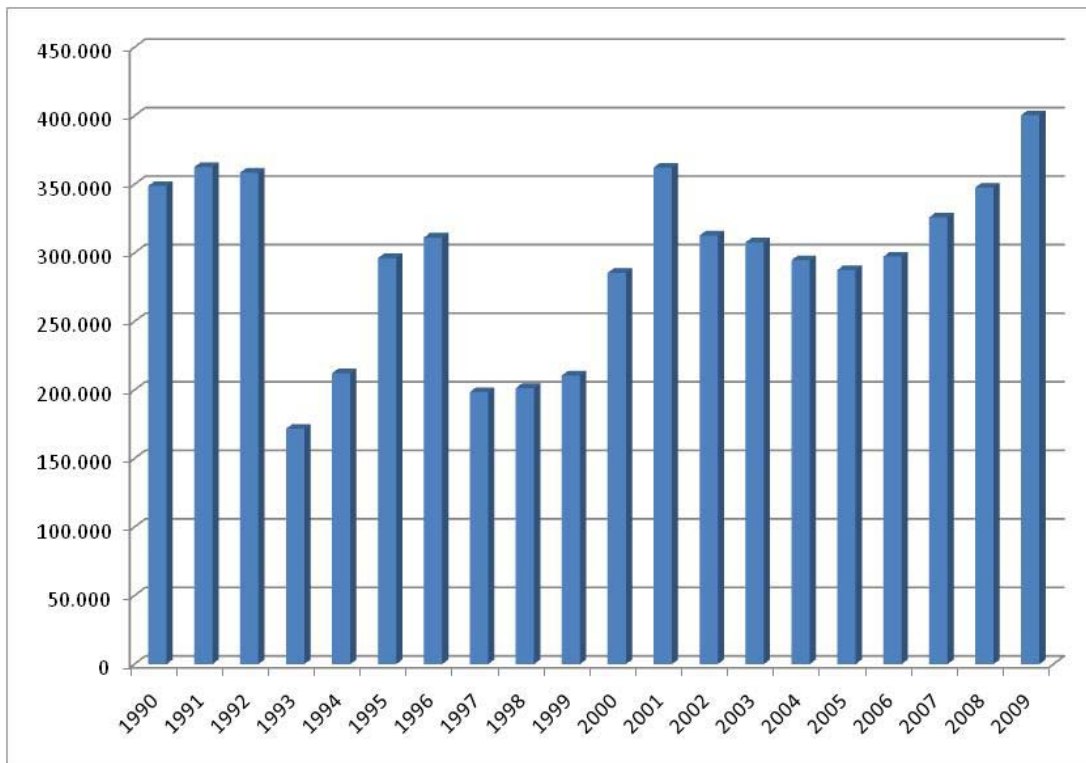
Durch diese Maßnahmen gewinnen die Regelungen zur Verlustverrechnung an Gewicht, da das Unternehmen weniger profitabel ist und der hohe Verlust in Periode 6 häufig dazu führt, dass in folgenden Perioden die Ertragsteuern auf null reduziert werden. 1998 tritt durch den Wegfall der Gewerbesteuer vom Kapital eine Entlastung ein.

Interessanter Weise hat die Einführung der sog. Mindestbesteuerung 2004 nur einen geringen Effekt. Ebenso wird die Beschränkung des Verlustrücktrags im Jahre 1999 durch die Senkung des Körperschaftsteuersatzes auf 40 % mehr als ausgeglichen. Der Verlauf dieser Steuersatzänderung ist nun nicht mehr ablesbar. Die Ergebnisse zeigen zum Teil sogar einen gegenläufigen Effekt.

Erstaunlich ist, dass sogar der Wegfall der Vermögensteuer 1997 und der Gewerbesteuer vom Kapital 1998 keine signifikante und langfristige Auswirkung hat und

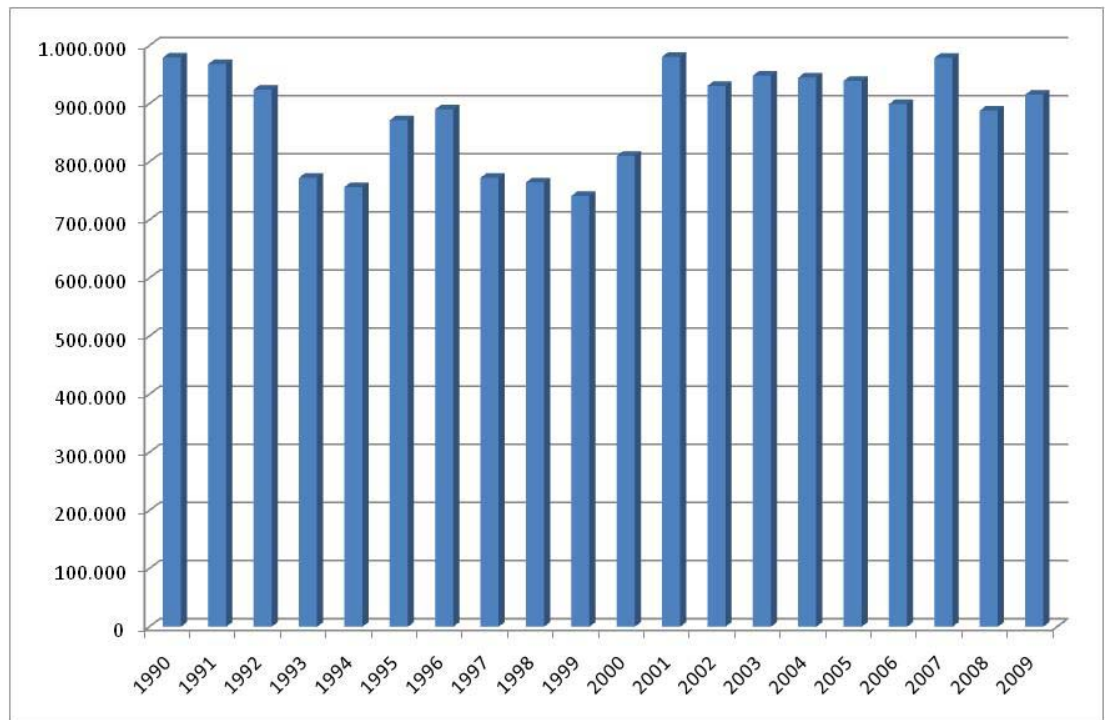
durch Änderungen an der Bemessungsgrundlage (z. B. Verschlechterung der Abschreibungsbedingungen 2001) kompensiert wird.

**Abbildung 118: Steuerbelastung in Deutschland auf Anteilseignerebene im Zeitablauf (Finanzkrise)**



Auf Ebene der Anteilseigner fällt die Aussetzung der Vermögensteuer 1997 im Vergleich zum ansonsten fast identischen Jahr 1996 ins Auge. Der Unterschied zwischen 1993 und 1994 beruht primär auf der reduzierten Anrechnung. Der Anstieg 2009 auf Anteilseignerebene ist primär der Abgeltungssteuer geschuldet, da nun Dividenden mit 25 % anstatt mit 21 % (42 % durch 2 - Halbeinkünfteverfahren) besteuert werden.

*Abbildung 119: Steuerbelastung in Deutschland auf Gesamtebene im Zeitablauf (Finanzkrise)*



## 6.11 Sicherung (Backup) der Datenbank

Zurzeit wird jeden Tag eine Kopie des Inhaltes der Datenbank, ein sog. Dump<sup>275</sup>, erstellt. Zudem können alle Länder in das Programm geladen und als XML exportiert werden. Dies kann auch als temporäres Backup benutzt werden.

Dieser Dualismus von zwei getrennten Speicherungsarten hat den zusätzlichen Vorteil, dass Reorganisationen möglich werden. Das Datenbankschema kann so z. B. geändert werden, solange die XML-Speicherung unverändert bleibt, da durch das Backup in der anderen Speicherungsart der Inhalt wieder problemlos rekonstruiert werden kann.

Dies ist prinzipiell auch umgekehrt möglich, aber nicht gewünscht, da die Speicherung außerhalb der Datenbank nur temporär erfolgen sollte.

Zudem kann der Administrator auch während der Laufzeit des Programms die Datenbankverbindung wechseln (vgl. Abbildung 120). Somit ist ein Umzug der Datenbank leicht möglich. Solch ein Transfer kann nur den Wechsel eines Rechners bedeu-

<sup>275</sup> Dies kann mittels „pgdump“ und „cron“ erfolgen.

ten (z. B. vom Server auf den Arbeitsplatz-PC und umgekehrt) oder gar den Wechsel des Datenbanksystems (z. B. der Wechsel von PostgreSQL<sup>276</sup> zu DB2<sup>277</sup>).

*Abbildung 120: Wechsel der Datenbankbindung*



Allerdings sind für einen Wechsel des Datenbank Managements-Systems noch einige weitere Modifikationen notwendig. Zum einen müssen für beide Systeme die Treiber für die Java Database Connectivity (JDBC) Schnittstelle eingebunden werden und zum anderen muss der Datenbank Dialekt in der Konfiguration Hibernate gewechselt werden. Den Dialekt könnte man zwar aus dem Uniform Resource Locator (URL) extrahieren, jedoch wird normalerweise nur ein JDBC-Treiber benötigt. Somit empfiehlt es sich in diesen wohl seltenen Ausnahmefällen, eine separate Version zu erstellen. Deswegen ist obiger Dialog nur für den einfachen Wechsel zwischen zwei Rechnern gedacht.

<sup>276</sup> Siehe <http://www.postgresql.org>.

<sup>277</sup> DB2 ist eine Datenbank von IBM. Vgl. <http://www-01.ibm.com/software/data/db2>.



## 7 Zusammenfassung

Ziel der Arbeit war es, die Infrastruktur für ein Archiv von steuerlichen Regelungen zu schaffen. Dieses sollte die aktuelle versionsabhängige Speicherung ablösen. Weitere zentrale Aspekte stellen die Verbesserung der Dokumentation und die zentralisierte Speicherung, sowie die Qualitätssicherung dar. Zum einen soll dadurch Wissen und Methodenkompetenz für alle Mitarbeiter zugänglich werden. Zum anderen wird so sukzessive die Datenbasis für Zeitreihenanalysen geschaffen.

Zur Sicherung der Qualität der modellierten Steuersysteme wurden eine Benutzerverwaltung und eine Versionshistorie eingeführt. Hierdurch ist es nun möglich, die Qualität der hinterlegten Steuerinformationen zu ermitteln und so eine erneute Validierung zu vermeiden. Änderungen werden nun dokumentiert und sind leichter nachvollziehbar. Ebenso wurde eine Kontrollfunktion zur Aufbereitung der hinterlegten Informationen über die Steuersysteme eingeführt und somit die Fehlerentdeckungswahrscheinlichkeit erhöht.

Des Weiteren schafft die klare Trennung zwischen aktuellem Rechtsstand und anderen Variationen, wie z. B. nicht umgesetzte, abgewandelte oder erst später realisierte Reformvorschläge die Basis, um Vermischungen verschiedener Varianten in Zukunft zu verhindern.

Zudem ist die kontextbezogene Hinterlegung der relevanten Dokumentation im Archiv nun möglich. Dadurch sind einmal recherchierte Informationen nun zentral und über Projektgrenzen hinweg zugreifbar. Durch die Recherche im Webinterface sind diese Informationen, losgelöst von der Kenntnis der genauen Funktionsweise des European Tax Analyzer, nutzbar.

Die Einführung des Verlustverrechnungsmoduls und weiterer Optionen (wie z. B. zusätzlicher Abschreibungsverläufe) erleichtert die Hinterlegung der kodifizierten Regelungen und vereinfacht die Modellierung. Durch all diese Maßnahmen wird die effiziente Berechnung von Zeitreihen zusätzlich unterstützt. Ein Nebeneffekt der Reorganisation des Datenmodells besteht darin, dass bereits vorhandene Benutzerdialoge optimiert, vereinheitlicht und vereinfacht werden konnten.

Die konsequente Umsetzung der Mehrbenutzerfähigkeit stellt die Grundlage für eine ausreichende Performanz und Verfügbarkeit dar. Des Weiteren hat die Wahl einer relationalen Datenbank die Zukunftssicherheit erhöht, da sie im Gegensatz zu einer

objektorientierten Datenbank nicht auf eine konkrete Programmiersprache festgelegt ist. Zudem ermöglicht die Wahl des Persistenzframeworks, dass das Datenbankmanagementsystem bei Bedarf auf einfache Art und Weise gewechselt werden kann, da Anbieter spezifische Elemente vermieden wurden. Somit ist eine Anpassung an zukünftig eventuell erhöhte Anforderungen ohne Weiteres möglich. Mithilfe von Annotationen bilden Quellcode und Datenbankschema eine Einheit, wodurch komplizierte Transformationsroutinen nicht benötigt werden.

Des Weiteren versetzt die Option der temporären Speicherung den Anwender in die Lage, das Programm auch ohne direkte Verbindung zur Datenbank zu nutzen. Zudem eröffnet sich hierdurch die Möglichkeit, eventuell notwendige Änderungen am Datenschema mit geringem Aufwand zu realisieren.

Der exemplarisch durchgeführte Anwendungsfall illustriert die Funktionsweise des Systems. Zudem wird die systemunabhängige Suche und Recherche von Steuerinformationen dargestellt. Darüber hinaus ist nun, durch die Hinterlegung von Abfragen, die Darstellung der zeitlichen Entwicklung von Steuertarifen oder die Aufbereitung anderer Teilaspekte möglich. Hierdurch ergeben sich neue Anwendungsaspekte, die den bisherigen Funktionsumfang deutlich erweitern und auch die Grundlage für weitere Entwicklungen und Effizienzgewinne bilden können.

## Literaturverzeichnis

- Bachfeld, D. (Giftspritze, 2004). "Giftspritze."  
<http://www.heise.de/security/Giftspritze--/artikel/43175>.
- Bachfeld, D. (Sicherheitslücken bleiben ungepatcht, 2009). "Studie: Viele Sicherheitslücken bleiben ungepatcht." <http://www.heise.de/security/Studie-Viele-Sicherheitsluecken-bleiben-ungepatcht--/news/meldung/126785>.
- Bauch, G. /Oestreicher, A. (1993): Handels- und Steuerbilanzen : einschliesslich der Systematik betrieblicher Ertrags- und Substanzsteuern und der Vermögensaufstellung ; mit Tabellen, 5., neubearb. und erw. Aufl., 14. - 18. Tsd.. Heidelberg 1993.
- Berger, R. F./Hase, A./Roock, S. /Sanitz, S. (Hibernate, 2006): Hibernate, Heidelberg 2006.
- Bloch, J. (Effective Java, 2008): Effective Java, Second Edition, 2008.
- Bundesministerium der Finanzen (2000). "AfA-Tabellen."  
[http://www.bundesfinanzministerium.de/nr\\_96040/DE/Wirtschaft\\_und\\_Verwaltung/Steuern/Veroeffentlichungen\\_zu\\_Steuerarten/Betriebspruefung/AfA-Tabellen/005.html](http://www.bundesfinanzministerium.de/nr_96040/DE/Wirtschaft_und_Verwaltung/Steuern/Veroeffentlichungen_zu_Steuerarten/Betriebspruefung/AfA-Tabellen/005.html), 18.07.2010.
- Collins-Sussman, B./Fitzpatrick, B. W. /Pilato, C. M. (Subversion, 2008): Version Control with Subversion, 2008.
- Deutsche Bundesbank (2003): Verhältniszahlen aus Jahresabschlüssen deutscher Unternehmen von 1998 bis 2000, Statistische Sonderveröffentlichung 6, Frankfurt a.M. 2003.
- Eckerle, T. H. (Standortunterschiede, 2000): Der Einfluß der Besteuerung auf die unternehmerische Investitionsentscheidung. Quantitative Belastungswirkungen im internationalen Vergleich, Lohmar -- Köln 2000.
- Engesser, H. /Claus, V. (1993): Duden, Informatik : ein Sachlexikon für Studium und Praxis, 2., vollst. überarb. und erw. Aufl.. Mannheim {[u.a.] 1993.
- Fowler, M. (Refactoring, 1999): Refactoring, Boston 1999.
- Gutekunst, G. (Steuerbelastungen, 2005): Steuerbelastungen und Steuerwirkungen bei nationaler und grenzüberschreitender Geschäftstätigkeit, Lohmar -- Köln 2005.
- Hennebrüder, S. (Hibernate Praxisbuch, 2007): Hibernate : das Praxisbuch für Entwickler, 1. Aufl.. Bonn 2007.
- IBFD. "IBFD." <http://www.ibfd.org/>, 10. 11. 2009.

- IBM (Cryptographic Coprocessor). "IBM PCI Cryptographic Coprocessor."  
<http://www-03.ibm.com/security/cryptocards/pcicc/overproduct.shtml>,  
 09. 02. 2009.
- Ihrig, H. /Pflaumer, P. (2009): Finanzmathematik Intensivkurs, 11. Auflage, München 2009.
- Jacobs, O. H. /Spengel, C. (European Tax Analyzer, 1996): European Tax Analyzer : EDV-gestützter Vergleich der Steuerbelastungen von Kapitalgesellschaften in Deutschland, Frankreich und Großbritannien, 1. Aufl.. Baden-Baden 1996.
- Kemper, A. /Eickler, A. (2009): Datenbanksysteme : eine Einführung, 7., aktualisierte und erweiterte Aufl.. München 2009.
- Mahmoud, Q. H. (2004). "Using and Programming Generics in J2SE 5.0."  
<http://java.sun.com/developer/technicalArticles/J2SE/generics/>, 17. 09. 2009.
- Meyer, R. (Simulation von Steuerbelastungen, 1996): Computergestützte Simulation von Steuerbelastungen, Band 12, Mannheim 1996.
- Milne, P. (XMLEncoder). "Using XMLEncoder."  
<http://java.sun.com/products/jfc/tsc/articles/persistence4/>, 09. 02. 2009.
- O. V. (Hibernate Reference Manual): Hibernate Reference Manual, 3.2.41 sp1.
- Oestreicher, A./Reister, T. /Christoph, S. (2009). "Common Corporate Tax Base (CCTB) and Effective Tax Burdens in the EU Member States." World Tax Journal (WTJ)
- Red Hat Middleware. "Open Session in View." <https://www.hibernate.org/43.html>, 08. 09. 2009.
- Reister, T. (Mikrosimulationsmodelle, 2009): Steuerwirkungsanalysen unter Verwendung von unternehmensbezogenen Mikrosimulationsmodellen, 1. Aufl.. Wiesbaden 2009.
- Sachverständigenrat zur Begutachtung der Gesamtwirtschaftlichen Entwicklung/Max-Planck-Institut /ZEW (Duale Einkommensteuer, 2006): Reform der Einkommens- und Unternehmensbesteuerung durch die duale Einkommensteuer : Expertise im Auftrag der Bundesminister der Finanzen und für Wirtschaft und Arbeit, Bonn 2006.
- Schadow, G. (Thick-Client Applications). "Best Practices for Thick-Client Applications (i.e., non-web apps.)." <https://www.hibernate.org/333.html>, 08. 09. 2009.
- Singular Systems (JEP). "<http://www.singularsys.com/jep/>."  
<http://www.singularsys.com/jep/>, 31. 8. 2009.

- Spengel, C. (Steuerbelastungsvergleiche, 1995): Europäische Steuerbelastungsvergleiche, Düsseldorf 1995.
- Spengel, C. (Taxation Knowledge Database, 2004): Taxation Knowledge Database, Mannheim 2004.
- Spengel, C./Finke, K. /Zinn, B. (Substanzbesteuerung, 2010): Bedeutung der Substanzbesteuerung in Deutschland - Eine quantitative Analyse unter Einbezug von Reformüberlegungen, Band 96, 2010.
- Stetter, T. (VITAX, 2005): Computergestützte internationale Steuerbelastungsvergleiche, Köln 2005.
- Sun Microsystems Inc. (Enums, 2004). "Enums."  
<http://java.sun.com/j2se/1.5.0/docs/guide/language/enums.html>, 29. 01. 2009.
- Trinkwalder, A. (PDF/A als ISO-Standard, 2005). "Langzeitarchivierung: PDF/A als ISO-Standard." <http://www.heise.de/newsticker/Langzeitarchivierung-PDF-A-als-ISO-Standard--/meldung/63957>.
- Wutka, M. (J2EE guide, 2002): J2EE developer's guide : [JSP, Servlets, EJB 2.0, JNDI, JMS, JDBC, Corba, XML, RMI], [Nachdr.]. München 2002.

# Curriculum Vitae

**Michael Grünewald**

Diplom-Wirtschaftsinformatiker

## Praxiserfahrung

Seit 09/2010	Angestellter beim <b>DWD</b> (Deutscher Wetterdienst)
01/05 – 08/10	Wissenschaftlicher Mitarbeiter am <b>ZEW</b> (Zentrum für europäische Wirtschaftsforschung) in Mannheim
5/01 – 8/01	Praktikum bei der Siemens AG Mannheim (IT-SAS)

## Hochschulstudium

3/03	Abschluss als Diplom-Wirtschaftsinformatiker
Thema der Diplomarbeit:	Template basierte Codeerzeugung für verteilte Systeme
Thema der Studienarbeit:	Network Aware Internet Video Encoding
9/97 - 2/98	Auslandssemester an der Napier University in Edinburgh/Schottland
10/94	Wirtschaftsinformatik an der Universität Mannheim

## Schulbildung

1985-1994	Heinrich Böll Gymnasium, Ludwigshafen
1982-1983	Neumayerschule in Frankenthal (Pfalz)
1981-1982 / 1983-1985	Grundschule in Ludwigshafen-Gartenstadt